



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**TUGAS AKHIR - TE 141599**

**IMPLEMENTASI DAN EVALUASI KINERJA *FREQUENCY HOPPING SPREAD SPECTRUM* (FHSS) MENGGUNAKAN WARP**

Rini Nurpita Sari  
NRP 2212106013

Dosen Pembimbing  
Dr. Ir. Suwadi, M.T.  
Dr. Ir. Titiek Suryani, M.T.

JURUSAN TEKNIK ELEKTRO  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2015



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**FINAL PROJECT - TE 141599**

**IMPLEMENTATION AND PERFORMANCE EVALUATION OF  
FREQUENCY HOPPING SPREAD SPECTRUM (FHSS) USING  
WARP**

Rini Nurpita Sari  
NRP 2212106013

Supervisor  
Dr. Ir. Suwadi, M.T.  
Dr. Ir. Titiek Suryani, M.T.

DEPARTMENT OF ELECTRICAL ENGINEERING  
Faculty of Industrial Technology  
Sepuluh Nopember Institute of Technology  
Surabaya 2015

**IMPLEMENTASI DAN EVALUASI KINERJA  
FREQUENCY HOPPING SPREAD SPECTRUM (FHSS)  
MENGGUNAKAN WARP**

**TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Sarjana Teknik  
Pada  
Bidang Studi Telekomunikasi Multimedia  
Jurusan Teknik Elektro  
Institut Teknologi Sepuluh Nopember**

**Menyetujui :**

**Dosen Pembimbing**

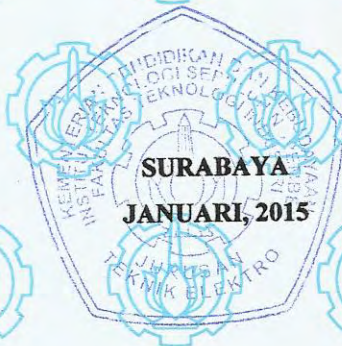


**Dr. Ir. Suwadi, M.T.**  
**NIP. 1968 0818 199303 1 002**

**Dosen Pembimbing II,**



**Dr. Ir. Titiek Suryani, MT.**  
**NIP. 1964 1130 198903 2 001**



# IMPLEMENTASI DAN EVALUASI KINERJA *FREQUENCY HOPPING SPREAD SPECTRUM* (FHSS) MENGGUNAKAN WARP

**Nama** : Rini Nurpita Sari  
**Pembimbing** : Dr. Ir. Suwadi, M.T.  
Dr. Ir. Titiek Suryani, M. T.

## ABSTRAK

Proses pengiriman informasi dari pengirim ke penerima terkadang terdapat gangguan, maka untuk mengatasi masalah tersebut dapat menggunakan metode *Spread Spectrum*, yang merupakan teknik pengiriman sinyal yang tahan terhadap gangguan berupa *noise* maupun *jamming*. *Frequency Hopping Spread Spectrum* merupakan sinyal informasi ditransmisikan yang dilompat ke dalam suatu spektrum frekuensi secara random atau sesuai pola tertentu. Spektrum tersebut sudah dialokasikan menjadi beberapa *channel* yang diperuntukkan untuk *frequency hopping signal*. Dalam Tugas Akhir ini, akan diimplementasikan sistem komunikasi SISO pada FHSS menggunakan *Wireless Open Access Research Platform* (WARP). Sistem FHSS ini akan dimodelkan dengan menggunakan *software* Matlab. Ketahanan sistem FHSS terhadap *Single Tone Jamming* dan *Multi Tone Jamming* akan diketahui dengan melakukan beberapa pengujian. Hasil yang diperoleh nilai *Bit Error Rate* (BER) akan semakin kecil jika daya yang dipancarkan semakin besar, sedangkan pada *jamming* dipengaruhi oleh nilai daya *jamming*. Penelitian ini menunjukkan bahwa kinerja sistem berjalan baik saat *Single Tone Jamming* lebih baik dari pada *Multi Tone Jamming*. Hal ini dibuktikan dengan hasil nilai BER saat daya *jamming* 300% untuk *Single Tone Jamming* sebesar 0.4194, dan *Multi Tone Jamming* sebesar 0.4892. Rata-rata sistem tidak terpengaruh oleh gangguan *jamming*, saat nilai daya *jamming* 75% dan 100%. Hasil penelitian ini menunjukkan bahwa sistem FHSS lebih tahan terhadap *Single Tone Jamming* dan *Multi Tone Jamming* dibandingkan sistem 4-ary FSK.

**Kata kunci** : *Frequency Hopping, Spread Spectrum, Jamming, Bit Error Rate, WARP, Matlab*



# **IMPLEMENTATION AND PERFORMANCE EVALUATION OF FREQUENCY HOPPING SPREAD SPECTRUM (FHSS) USING WARP**

**Nama** : Rini Nurpita Sari  
**Pembimbing** : Dr. Ir. Suwadi, M.T.  
Dr. Ir. Titiek Suryani, M. T.

## **ABSTRACT**

*The process of sending an information from transceiver to receiver distortions come occasionally. to overcome those problems, the spread spectrum method could be used, it is technique to transmit signal that suffer from distortion such as noise and jamming. Frequency Hopping Spread Spectrum is transmitted information signal which jumped to one of frequency spectrum randomly or based on certain path. Those spectrum already allocated in to some channel for Frequency Hopping Signal. In this paper, an implementation of SISO communication system on FHSS system Using Wireless Open Access Research Platform (WARP) would be done. This FHSS system will be modeled using matlab software. Immunity of FHSS system to Single Tone Jamming and Multi Tone Jamming will be known by implementing some test. The result implying that Bit Error Rate (BER) value will smaller if the transmitting power larger While on jamming is influenced by jamming power. This research show that the system performance going smooth on Single Tone Jamming compared to Multi Tone Jamming. It is proved with the result of BER value when the jamming power is 300% for single tone jamming is 0,4194, and multi tone jamming is 0,4892. Commonly the system is not influenced by jamming distortion, when the jamming power is 75% and 100%. This research result shows that FHSS system more endure to single tone jamming and multi tone jamming compared to any 4-ary FSK system.*

**Keywords** : Frequency Hopping, Spread Spectrum, Jamming, Bit Error Rate, WARP, Matlab

## KATA PENGANTAR

Alhamdulillahirabbil'aalamiin, segala puji syukur penulis panjatkan kepada Allah SWT atas segala rahmat dan hidayah yang telah diberikan-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul :

### **“Implementasi dan Evaluasi Kinerja *Frequency Hopping Spread Spectrum* (FHSS) Menggunakan WARP”**

Tugas Akhir ini dibuat untuk memenuhi persyaratan mendapatkan gelar Strata I pada jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Penulis menyadari bahwa buku ini jauh dari sempurna, oleh sebab itu saran dan kritik yang bersifat membangun senantiasa dinantikan. Penulis juga berharap agar Tugas Akhir ini dapat dikembangkan lebih lanjut sehingga dapat digunakan sebaik-baiknya untuk mendukung perkembangan ilmu pengetahuan.

Dalam kesempatan ini penulis menyampaikan penghargaan dan rasa terima kasih sebesar-besarnya kepada semua pihak yang telah mendukung serta membantu penulis dalam menyelesaikan Tugas Akhir ini, khususnya kepada:

1. Ibu dan Adik saya, Ibu Nurkayah dan Adik Yanti Rahayu yang selalu memberikan dukungan, semangat, doa, dan memberikan bantuan materil dan non-materil.
2. Bapak Dr. Ir. Suwadi, M.T dan Dr. Ir. Titiek Suryani, M.T selaku Dosen Pembimbing atas segala bantuan, perhatian, dan arahan selama pengerjaan Tugas Akhir ini.
3. Teman-teman kelas TMM LJ 2012 Genap dan seluruh warga Laboratorium Komunikasi Multimedia B304.
4. Semua keluarga dan rekan-rekan yang banyak terlibat dalam pengerjaan tugas akhir ini dan tidak bisa disebutkan satu persatu oleh penulis.

Surabaya, Januari 2015

Penulis

## DAFTAR ISI

	Halaman
<b>PERNYATAAN KEASLIAN TUGAS AKHIR.....</b>	<b>i</b>
<b>ABSTRAK .....</b>	<b>v</b>
<b>ABSTRACT .....</b>	<b>vii</b>
<b>KATA PENGANTAR.....</b>	<b>ix</b>
<b>DAFTAR ISI.....</b>	<b>xi</b>
<b>DAFTAR GAMBAR.....</b>	<b>xv</b>
<b>DAFTAR TABEL .....</b>	<b>xvii</b>
 <b>BAB I PENDAHULUAN .....</b>	 <b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	3
1.5 Metodologi .....	3
1.6 Sistematika Penulisan.....	4
1.7 Relevansi atau Manfaat .....	5
 <b>BAB II TEORI PENUNJANG.....</b>	 <b>7</b>
2.1 Sistem Komunikasi Nirkabel .....	7
2.1.1 Karakteristik Kanal Sistem Komunikasi Nirkabel .....	7
2.2 <i>Frequency Hopping Spread Spectrum</i> (FHSS) .....	8
2.2.1 Konsep dan Karakteristik FHSS .....	9
2.3 Modulasi MFSK.....	13
2.4 Teknik <i>Jamming</i> .....	14
2.4.1 <i>Single Tone Jamming</i> (STJ).....	14
2.4.2 <i>Multiple Tone Jamming</i> (MTJ) .....	15
2.5 <i>Software Defined Radio</i> (SDR).....	16
2.6 <i>Wireless Open-Access Research Platform</i> (WARP) .....	17
2.6.1 <i>Arsitektur Platform</i> .....	17
2.6.1.1 Desain Hardware.....	19
2.6.1.2 WARP Design Flows.....	23
2.6.1.3 Open-Access Repository.....	23
2.6.2 WARPLab.....	24
 <b>BAB III PERANCANGAN DAN IMPLEMENTASI SISTEM .....</b>	 <b>27</b>



3.1	Pendahuluan.....	27
3.2	Integrasi <i>Personal Computer</i> (PC) dengan Modul WARP.....	29
3.2.1	Integrasi Sistem FHSS Tanpa <i>jamming</i> .....	29
3.2.2	Integrasi Sistem FHSS dengan <i>Jamming</i> .....	32
3.3	Skenario Penelitian.....	35
3.3.1	Skenario Pengukuran.....	35
3.4	Desain Sistem Komunikasi FHSS pada WARP .....	37
3.4.1	Pembangkitan Bit Informasi.....	39
3.4.2	Proses Modulasi.....	39
3.4.3	Pola Lompatan.....	40
3.4.4	<i>Inverse fast Fourier transform</i> (IFFT).....	40
3.4.5	Interpolation .....	41
3.4.6	Pengiriman Melalui Modul WARP.....	43
3.4.7	Proses di Penerima.....	43
3.4.7.1	<i>Decimate</i> .....	43
3.4.7.2	<i>LTS Correlation</i> .....	44
3.4.7.3	<i>Fast Fourier Transform</i> (FFT) .....	45
3.4.7.4	<i>Equalization</i> .....	45
3.4.7.5	Demodulator 4-ary FSK.....	47
3.5	Desain Sistem Komunikasi FHSS dengan <i>Jamming</i> pada WARP.....	47
3.6	Perhitungan BER.....	47
3.7	Metode Pengukuran.....	48
<b>BAB IV PENGUJIAN DAN ANALISIS DATA .....</b>		<b>51</b>
4.1	Analisis Kinerja Implementasi Sistem tanpa <i>Jamming</i> .....	51
4.2	Analisis Kinerja Implementasi FHSS dengan Gangguan <i>Jamming</i> .....	55
4.2.1	Frekuensi Sinyal <i>Jamming</i> sama dengan Frekuensi Kerja pada Gangguan <i>Single Tone Jamming</i> .....	56
4.2.2	Frekuensi Sinyal <i>Jamming</i> Tidak sama dengan Frekuensi Kerja pada pengujian <i>Single Tone Jamming</i> .....	57
4.2.3	Frekuensi Sinyal <i>Jamming</i> sama dengan Frekuensi Kerja pada pengujian <i>Multi Tone Jamming</i> .....	60
4.2.4	Frekuensi Sinyal <i>Jamming</i> Tidak sama dengan Frekuensi Kerja pada pengujian <i>Multi Tone Jamming</i> .....	61
4.3	Analisis Sistem .....	64
<b>BAB V PENUTUP.....</b>		<b>71</b>
5.1	Kesimpulan .....	71
5.2	Saran.....	72



<b>DAFTAR PUSTAKA .....</b>	<b>73</b>
<b>LAMPIRAN A PROPOSAL TUGAS AKHIR.....</b>	<b>75</b>
<b>LAMPIRAN B LEMBAR MONITORING.....</b>	<b>87</b>
<b>LAMPIRAN C LISTING PROGRAM.....</b>	<b>89</b>
<b>LAMPIRAN D SPEKTRUM <i>JAMMING</i>.....</b>	<b>99</b>
<b>LAMPIRAN E PENGUJIAN MENGGUNAKAN SPEKTRUM ANALYZER.....</b>	<b>101</b>
<b>RIWAYAT HIDUP PENULIS .....</b>	<b>103</b>

## DAFTAR GAMBAR

GAMBAR	HALAMAN
Gambar 2.1 <i>Co-located Frequency Hopping Systems</i> .....	9
Gambar 2.2 <i>Pola Frequency-hopping</i> .....	9
Gambar 2.3 <i>Frequency Plot dari Fast frequency hopping</i> .....	10
Gambar 2.4 <i>Frekuensi Plot dari Slow frequency hopping</i> .....	11
Gambar 2.5 <i>Single Tone Jamming</i> .....	14
Gambar 2.6 <i>Multiple Tone Jamming</i> .....	15
Gambar 2.7 Empat komponen utama dalam WARP .....	17
Gambar 2.8 FPGA Board .....	20
Gambar 2.9 Revisi Pertama dari <i>Platform Custom Hardware</i> , termasuk <i>Board Virtex-II Pro FPGA, Radio Daughtercard</i> dan <i>Video Daughtercard</i> .....	21
Gambar 2.10 <i>WARP Design Flows</i> .....	23
Gambar 2.11 Gambar WARPLab .....	25
Gambar 2.12 <i>Setup WARPLab</i> .....	25
Gambar 3.1 Blok Diagram Langkah-Langkah Penelitian .....	28
Gambar 3.2 Konfigurasi Sistem Komunikasi SISO Pada WARP .....	29
Gambar 3.3 Pengaturan IP Address FHSS tanpa <i>Jamming</i> .....	30
Gambar 3.4 Cmd FHSS tanpa <i>Jamming</i> .....	30
Gambar 3.5 Kondisi Ping antara <i>Node 3</i> dan <i>Node 4</i> .....	31
Gambar 3.6 Indikator Keberhasilan Koneksi WARP dengan PC .....	32
Gambar 3.7 Konfigurasi Sistem FHSS dengan <i>Jamming</i> .....	32
Gambar 3.8 Pengaturan IP Address FHSS untuk <i>Jamming</i> .....	33
Gambar 3.9 Cmd untuk <i>Jamming</i> .....	33
Gambar 3.10 Kondisi Ping <i>Node</i> untuk <i>Jamming</i> .....	34
Gambar 3.11 Indikator Koneksi WARP dengan PC untuk <i>Jamming</i> ..	34
Gambar 3.12 Skenario Pengukuran .....	35
Gambar 3.13 Tx Output Power vs Gain Setting pada MAX2829 .....	36
Gambar 3.14 Gain RX <i>Baseband</i> pada MAX2829 .....	37
Gambar 3.15 Pemodelan Sistem Komunikasi dengan Teknik FHSS ...	38
Gambar 3.16 <i>Frequency Hopping Spread Spectrum (FHSS)</i> .....	40
Gambar 3.17 Urutan <i>Subcarrier Data</i> dan <i>Subcarrier Pilot</i> .....	41
Gambar 3.18 Proses Interpolation .....	42
Gambar 3.19 Hasil Plot Blok <i>Interpolation</i> .....	42
Gambar 3.20 Proses <i>Decimate</i> .....	43
Gambar 3.21 Hasil Plot Blok <i>Decimate</i> .....	44
Gambar 3.22 <i>LTS Correlation</i> .....	45

Gambar 3.23	Blok Diagram <i>Jamming</i> (a) <i>Single Tone</i> (b) <i>Multi Tone</i>	47
Gambar 3.24	(a) Bit yang Dikirim (b) Bit yang Diterima	48
Gambar 3.25	Pengukuran Tanpa <i>Jamming</i>	49
Gambar 3.26	Pengukuran Menggunakan <i>Jamming</i>	49
Gambar 4.1	Grafik BER vs Daya Pancar (dBm) untuk Sistem FHSS dan 4-ary FSK tanpa <i>Jamming</i>	53
Gambar 4.2	Grafik BER vs Daya Sinyal <i>Jamming</i> pada Frekuensi kerja sama dengan Frekuensi <i>jamming</i> saat <i>Single Tone Jamming</i>	57
Gambar 4.3	Grafik BER vs Daya Sinyal <i>Jamming</i> pada Frekuensi kerja tidak sama dengan Frekuensi <i>jamming</i> saat <i>Single Tone Jamming</i>	59
Gambar 4.4	Grafik BER vs Daya Sinyal <i>Jamming</i> pada Frekuensi kerja sama dengan Frekuensi <i>jamming</i> saat <i>Multi Tone Jamming</i>	61
Gambar 4.5	Grafik BER vs Daya Sinyal <i>Jamming</i> pada Frekuensi kerja tidak sama dengan Frekuensi <i>jamming</i> saat <i>Multi Tone Jamming</i>	63
Gambar 4.6	Grafik BER vs Daya Sinyal <i>Jamming</i> pada Gangguan <i>Single Tone Jamming</i> saat $P_{tx} = -19$ dBm	64
Gambar 4.7	Grafik BER vs Daya Sinyal <i>Jamming</i> pada Gangguan <i>Single Tone Jamming</i> saat $P_{tx} = -23$ dBm	66
Gambar 4.8	Grafik BER vs Daya Sinyal <i>Jamming</i> pada <i>Multi Tone Jamming</i> saat $P_{tx} = -19$ dBm	67
Gambar 4.9	Grafik BER vs Daya Sinyal <i>Jamming</i> pada <i>Multi Tone Jamming</i> saat $P_{tx} = -23$ dBm	68

## DAFTAR TABEL

TABEL	HALAMAN
Tabel 3.1	Parameter Sistem komunikasi SISO dengan Teknik FHSS 39
Tabel 4.1	Hasil Pengujian FHSS dan 4-ary FSK tanpa <i>jamming</i> ..... 52
Tabel 4.2	Nilai BER vs Daya <i>Single Tone Jamming</i> untuk Frekuensi <i>jamming</i> sama dengan Frekuensi kerja..... 56
Tabel 4.3	Nilai BER vs Daya <i>Single Tone Jamming</i> untuk Frekuensi <i>jamming</i> tidak sama dengan Frekuensi kerja ..... 58
Tabel 4.4	Nilai BER vs Daya <i>Multi Tone Jamming</i> untuk Frekuensi <i>jamming</i> sama dengan Frekuensi kerja..... 60
Tabel 4.5	Nilai BER vs Daya <i>Multi Tone Jamming</i> untuk Frekuensi <i>jamming</i> tidak sama dengan Frekuensi kerja ..... 62



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Semakin pesatnya perkembangan teknologi komunikasi, sehingga sekarang banyak peralatan komunikasi yang menggunakan media nirkabel yang memiliki banyak keunggulan, antara lain praktis dan mempunyai mobilitas yang tinggi. Tetapi pada implementasi sistem komunikasi nirkabel banyak mengalami gangguan yang dapat menurunkan kinerja sistem, kemudian kondisi kanal akan menurun yang mengakibatkan kapasitas kanal dan *reliability* dari data yang dikirimkan juga menurun. Gangguan pada sistem komunikasi nirkabel yang paling dominan adalah berupa *jamming*. Untuk mengatasi *jamming* tersebut dapat menggunakan sistem *Spread Spectrum*.

Dalam proses pengiriman informasi dari pengirim ke penerima terkadang terdapat gangguan, maka untuk mengatasi masalah tersebut dapat menggunakan metode *Spread Spectrum* yang merupakan salah satu jenis teknik pengiriman sinyal yang tahan terhadap gangguan berupa *noise* maupun *jamming*. Pada saat proses melakukan pengiriman informasi, teknik ini mempunyai tingkat keamanan informasi yang cukup tinggi, salah satu bagian dari *Spread Spectrum* adalah komunikasi *Frequency-Hopping Spread Spectrum* (FHSS) yang merupakan teknik modulasi sinyal informasi dengan sinyal pembawa (*carrier*) mengubah frekuensi secara acak tetapi sesuai urutan dan memiliki ketahanan yang tinggi terhadap *Single Tone Jamming* (STJ) dan *Multi Tone Jamming* (MTJ). Frekuensi yang berubah-ubah ini dipilih oleh kode-kode tertentu, agar dapat mengurangi kerusakan yang ditimbulkan oleh interferensi *noise* maupun *jamming* dan menjamin kerahasiaan sinyal yang dikirimkan.

Proses menyampaikan informasi dari pengirim ke penerima yang penting dalam alat telekomunikasi berupa radio, radio mampu menyesuaikan standar telekomunikasi setiap Negara dan tanpa merubah komponen radio tersebut untuk perkembangan aplikasi kapan saja. Aplikasi yang dapat mengatasi masalah tersebut adalah Software Defined Radio (SDR). Salah satu platform SDR yang sedang dikembangkan oleh Rice University adalah Wireless Open Access Research Platform (WARP). WARP merupakan desain platform open access tingkat tinggi yang sangat mampu, scalable, dan extensible untuk

penelitian sistem telekomunikasi nirkabel. Platform ini memiliki banyak keunggulan, salah satunya adalah WARP dibuat dengan disain perangkat keras khusus, mengintegrasikan sumber daya pengolahan berbasis Field Programmable Gate Array (FPGA) dengan antar muka radio nyata. Selain itu, platform ini juga didukung dengan modul khusus yang memudahkan pengguna dari berbagai pengolahan hardware dan sumber daya peripheral, dan platform ini juga mendukung modul-modul digunakan untuk membangun berbagai aplikasi penelitian, termasuk implementasi secara real time dari physical layer dan MAC layer.

Berdasarkan penjelasan diatas, maka dari itu penulis akan mengimplementasikan sistem komunikasi Frequency Hopping Spread Spectrum (FHSS) menggunakan Wireless Open Access Research Platform (WARP). Implementasi sistem FHSS menggunakan WARP ini penulis menerapkan dengan sistem komunikasi Single Input Single Output (SISO). Sistem komunikasi SISO memiliki kelebihan yaitu sedikit terjadi interferensi sinyal, dan mudah diterapkan karena hanya membutuhkan satu antenna sebagai pengirim dan satu sebagai penerima. Sistem komunikasi SISO, hanya membawa satu deretan data karena hanya terdapat satu kanal. Pada penelitian ini akan ditunjukkan kinerja sistem FHSS pada lingkungan indoor yang akan dilakukan di Gedung B304 Jurusan Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya.

## **1.2 Rumusan Masalah**

Permasalahan yang akan dibahas pada tugas akhir ini adalah sebagai berikut:

1. Bagaimana mensimulasikan pemancar dan penerima, dan *Frequency-Hopping Spread Spectrum* (FHSS) menggunakan M-file Matlab 2012a?
2. Bagaimana menguji ketahanan FHSS terhadap *jamming* ?
3. Bagaimana mengimplementasikan proses pengiriman dan penerimaan FHSS pada *Wireless Open-Access Research Platfrom* (WARP)?

## **1.3 Batasan Masalah**

Dalam penyelesaian Tugas Akhir ini, memiliki beberapa batasan masalah adalah sebagai berikut :

1. Pemodelan sistem pemancar dan penerima FHSS menggunakan perangkat lunak Matlab.

2. Menggunakan modul *Wireless Open-Access Research Platform* (WARP) untuk implementasi pemancar dan penerimaan FHSS.
3. Menguji jenis-jenis *jamming* yang akan diujikan terhadap *Frequency-Hopping Spread Spectrum* (FHSS) pada WARP.
4. Membandingkan hasil pengukuran dengan 4-ary FSK tanpa *jamming* maupun menggunakan *jamming*.
5. Pengujian data pada Tugas Akhir ini dilakukan di ruang B.304 Laboratorium Telekomunikasi Multimedia Jurusan Teknik Elektro.

#### 1.4 Tujuan

Tujuan dari penelitian Tugas Akhir ini, diharapkan dapat tercapainya suatu tujuan sebagai berikut :

1. Mengetahui kualitas kerja *Frequency-Hopping Spread Spectrum* (FHSS) jika diimplementasikan dengan menggunakan WARP.
2. Mengetahui kinerja *Frequency-Hopping Spread Spectrum* (FHSS) dengan gangguan *Single Tone Jamming* dan *Multi Tone Jamming* menggunakan WARP secara real.
3. Implementasi FHSS pada WARP sehingga dapat dimanfaatkan oleh mahasiswa sebagai panduan dan alat praktek pada Laboratorium B304.

#### 1.5 Metodologi

Metodologi pada Tugas Akhir ini adalah sebagai berikut :

1. Studi literatur  
Mencari referensi yang berkaitan dengan pemancar dan penerima FHSS dan implementasinya pada perangkat *Wireless Open-Access Research Platform* (WARP). Literatur yang digunakan meliputi buku referensi, jurnal dan dokumen internet.
2. Perancangan sistem pemancar dan penerima FHSS.  
Pemodelan sistem yang akan disimulasikan, meliputi perancangan sistem pemancar dan penerima FHSS dengan menggunakan perangkat lunak MatLAB.
3. Integrasi MatLaB pada *Wireless Open-Access Research Platform* (WARP).  
Proses integrasi model sistem yang telah dibuat dari MatLAB ke *Wireless Open-Access Research Platform* (WARP).



4. Pengujian sistem.  
Merupakan cara atau metode yang akan digunakan dalam pengujian sistem yang telah dibuat.
5. Analisa hasil keluaran sistem.  
Meliputi analisa hasil dari keluaran sistem pada MatLaB dibandingkan dengan hasil implementasi pada *Wireless Open-Access Research Platfrom* (WARP).
6. Kesimpulan.  
Penarikan kesimpulan dari kinerja sistem yang selanjutnya dapat ditarik sebuah kesimpulan.

## 1.6 Sistematika Penulisan

Sistematika penulisan pada Tugas Akhir ini dibagi menjadi beberapa bab dengan rincian sebagai berikut:

### BAB I : PENDAHULUAN

Pada bab ini menguraikan hal-hal berikut diantaranya latar belakang, perumusan masalah, batasan masalah, tujuan yang berkaitan dengan pengerjaan dan penyusunan Tugas Akhir ini, sistematika penulisan, metodologi dan relevansi atau manfaat.

### BAB II : TEORI PENUNJANG

Pada bab ini dikemukakan berbagai macam dasar teori yang berhubungan dengan permasalahan yang dibahas, antara lain meliputi teori tentang sistem pemancar dan penerima FHSS, konsep dari teknik *jamming* dan konsep *Wireless Open-Access Research Platform* (WARP).

### BAB III : PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini, berisi tentang tahap-tahap perancangan sistem pemancar dan penerima FHSS, integrasi *software* dan *hardware*, dan implementasi sistem dalam *Wireless Open-Access Research Platform* (WARP).

### BAB IV : PENGUJIAN DAN ANALISA DATA

Bab ini membahas mengenai hasil pengujian dari sistem yang telah diimplementasikan dan analisa data berdasarkan parameter yang ditetapkan.



## BAB V : PENUTUP

Berisi tentang kesimpulan hasil yang telah diperoleh dan saran yang terdapat dalam Tugas Akhir ini.

### 1.7 Relevansi atau Manfaat

Dalam Tugas Akhir ini diharapkan dapat menghasilkan suatu sistem pemancar dan penerima FHSS yang diimplementasikan pada *Wireless Open-Access Research Platform* (WARP) sehingga nantinya WARP tersebut dapat digunakan dalam sistem komunikasi *frequency hopping spread spectrum* (FHSS).

## **BAB II**

### **TEORI PENUNJANG**

#### **2.1 Sistem Komunikasi Nirkabel**

Komunikasi nirkabel merupakan komunikasi tanpa penghantar listrik atau tanpa kabel yang mentransfer informasi antara dua atau lebih titik. Dimana jarak antara penerima dan pengirim informasi bisa pendek, seperti beberapa meter untuk remote control televisi, atau sejauh ribuan bahkan jutaan kilometer seperti pada komunikasi radio. Pada komunikasi nirkabel ini melakukan telekomunikasi dengan menggunakan gelombang elektromagnetik sebagai media perantara pengganti kabel. Komunikasi nirkabel atau *wireless* saat ini semakin berkembang pesat, seperti perkembangan teknologi *wireless* yang digunakan untuk akses internet. Teknologi *wireless* memanfaatkan frekuensi tinggi untuk menghantarkan sebuah komunikasi, maka kerentanan terhadap keamanan juga lebih tinggi dibandingkan dengan teknologi komunikasi yang lainnya. Data ditransmisikan di frekuensi 2.4Ghz atau 5GHz. Contoh lain dari teknologi *wireless* adalah *headset*, penerima radio, siaran televisi tanpa kabel, telepon selular, dan *wireless mouse* [1].

Mekanisme telekomunikasi nirkabel memungkinkan komunikasi jarak jauh dengan lokasi yang tidak mungkin atau relatif sulit untuk dihubungkan dengan kabel, misalnya dalam bentuk pemancar atau penerima radio, pengendalian jarak jauh, jaringan komputer nirkabel. Umumnya telekomunikasi nirkabel menggunakan sarana gelombang elektromagnetik (misalnya laser, *Radio Frequency* (RF), cahaya) atau gelombang suara untuk mentransfer informasi tanpa menggunakan kabel.

##### **2.1.1 Karakteristik Kanal Sistem Komunikasi Nirkabel**

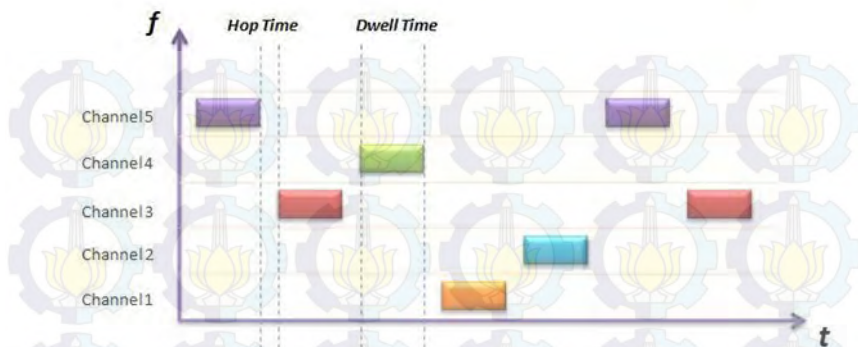
Jarak transmisi antara pemancar dan penerima dapat bervariasi dari *line of sight* (LOS) hingga tertutup oleh penghalang seperti bangunan, bukit dan pepohonan. Dimana kanal radio bersifat random dan tidak bisa di analisis dengan mudah, bahkan kecepatan dari pergerakan pengguna mempengaruhi seberapa cepat level sinyal memudar. Memodelkan kanal radio merupakan bagian yang paling sulit dalam mendesain sistem radio *mobile*, dan biasanya dilakukan dalam bentuk statistic yang berdasarkan atas hasil pengukuran [2].

Pada komunikasi nirkabel sinyal yang dikirimkan melalui media nirkabel akan mengalami atenuasi yang sangat besar, delay dan distorsi sinyal dibandingkan dengan melalui media kabel. Efek non-ideal tersebut pada umumnya disebabkan oleh tiga faktor yaitu, Path Loss, Shadowing dan multi-path fading. Komunikasi Path Loss misalnya terdapat sebuah sistem komunikasi point-to-point dimana sebuah pemancar berkomunikasi dengan penerima melalui pengiriman sinyal elektromagnetik melalui udara. Daya sinyal yang dikirimkan akan berkurang sepanjang sinyal tersebut berpropagasi melalui medium dan melemah semakin besar jarak yang ditempuh oleh sinyal.

## **2.2 Frequency Hopping Spread Spectrum (FHSS)**

*Frequency Hopping Spread Spectrum* (FHSS) merupakan sinyal informasi ditransmisikan yang disebar secara random ke beberapa frekuensi radio. Sinyal informasi tersebut dilampatkan ke dalam suatu spektrum frekuensi, dimana spektrum tersebut sudah dialokasikan menjadi beberapa *channel* yang diperuntukkan untuk *frequency hopping signal*. Lompatan sinyal informasi tersebut dapat dilakukan secara random ataupun menurut aturan tertentu. Formasi dari lompatan tersebut diatur berdasarkan kode tertentu, yang dinamakan *chipping code*, artinya termasuk bagian dari *encrypting signal*. Suatu spektrum atau rentang frekuensi akan disekat atau dibagi menjadi *channel-channel* teralokasi yang akan digunakan untuk melampatkan sinyal informasi dalam FHSS. Pembagian *channel-channel* tersebut didasarkan atas interval yang besarnya tetap, sehingga sinyal informasi akan dilampatkan dari satu frekuensi ke frekuensi lainnya dengan interval frekuensi yang tetap. Jarak interval tersebut dipengaruhi lebar dari spektrum yang digunakan untuk FHSS dan juga dipengaruhi oleh besarnya suatu sinyal informasi. Lebar dari beberapa *channel* yang digunakan sebagai lompatan dalam FHSS juga akan tergantung dari besarnya *bandwidth* dan input sinyal informasi. Apabila input sinyal informasi mempunyai *bandwidth* yang besar, maka otomatis lebar dari masing-masing *channel* sebagai tempat lompatannya juga akan besar, begitu pula sebaliknya.

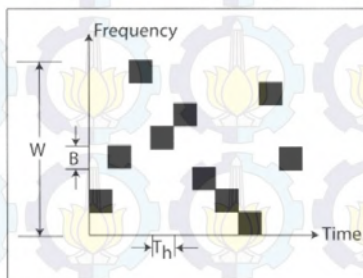




**Gambar 2.1** *Co-located Frequency Hopping Systems* [3]

### 2.2.1 Konsep dan Karakteristik FHSS

Pada *Frequency Hopping Spread Spectrum* (FHSS) urutan frekuensi pembawa disebut *Frequency-hopping pattern*. Himpunan frekuensi pembawa  $M \{f_1, f_2, \dots, f_M\}$  disebut *hopset*. Dimana perubahan frekuensi pembawa disebut *hop rate*. Lompatan yang terjadi melalui pita frekuensi disebut *hopping band* yang mencakup kanal frekuensi  $M$ . Setiap kanal frekuensi didefinisikan sebagai daerah spektral yang mencakup frekuensi pembawa tunggal dari *hopset* sebagai pusat frekuensi dan memiliki *bandwidth* cukup besar untuk mencakup daya dalam sinyal pulsa dengan spesifik frekuensi pembawa. Gambar 2.2 ilustrasi kanal frekuensi dikaitkan dengan keterangan *frequency-hopping pattern*. Interval waktu antara hop disebut *hop interval*. Durasinya disebut *hop duration* dan dilambangkan dengan  $T_h$ . *Hopping band* memiliki *bandwidth*  $W \geq MB$  [4].

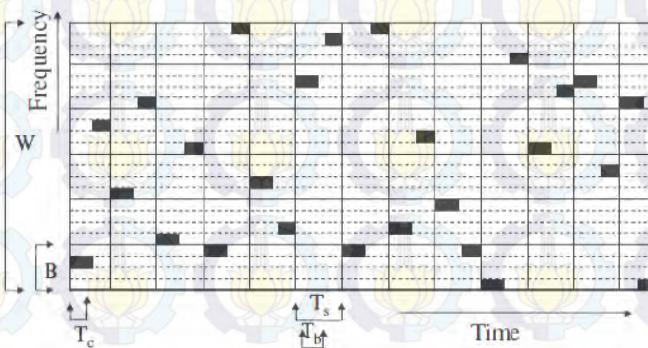


**Gambar 2.2** Pola *Frequency-hopping* [4]



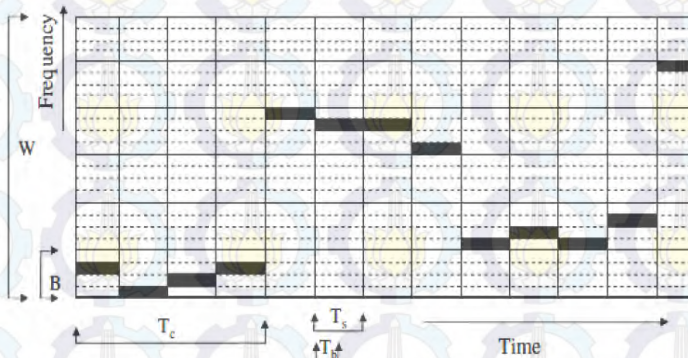
Frekuensi *hopping* tidak memberikan keunggulan terhadap *white noise*, frekuensi *hopping* memungkinkan sinyal untuk melompat keluar dari kanal frekuensi dengan interferensi atau *slow frequency-selective fading*. Untuk sepenuhnya memanfaatkan kemampuan frekuensi *hopping* terhadap sinyal interferensi *narrowband*, dibutuhkan kanal frekuensi yang dipisah-pisah. Pemisahan kanal mungkin berdekatan atau memiliki daerah spektral yang tidak terpakai diantara frekuensi *hopping*. Beberapa daerah spektral dengan interferensi yang stabil atau rentan terhadap *fading* dapat dihilangkan dari *hopset*, proses tersebut dinamakan *spectral notching*. *Multiple frequency-shift keying* (MFSK) secara fundamental berbeda dari frekuensi *hopping* bahwa semua sub-kanal MFSK mempengaruhi setiap keputusan disisi penerima. Maka dari itu tidak ada keluar atau menghindari sub-kanal dengan interferensi yang mungkin terjadi [4].

Frekuensi *hopping* dapat diklasifikasikan sebagai *fast* dan *slow*. *Fast frequency hopping* (FFH) terjadi jika ada lebih dari satu hop untuk setiap simbol informasi. Meskipun definisi ini tidak mengacu pada *hop rate*, *Fast frequency hopping* (FFH) merupakan pilihan hanya jika *hop rate* yang melebihi *information-symbol rate* dapat diimplementasikan. Pada *Fast frequency hopping* (FFH) nilai  $T_c < T_s$  yaitu frekuensi hopping terjadi lebih cepat dari modulasi. Sistem *fast* FH/MFSK berbeda dari sistem *slow* FH/MFSK bahwa ada beberapa lompatan setiap simbol M-ary. Oleh karena itu, dapat diterapkan dari *Fast frequency hopping* (FFH) yaitu  $T_s = NT_c$ . Gambaran dari sistem *fast* FHSS dengan modulasi 4-FSK ( $M = 4$ ), 6 frekuensi lompatan ( $L = 6$ ), dan 2 lompatan setiap simbol ( $T_s = 2T_c$ ,  $N = 2$ ). [5]



**Gambar 2.3** Frequency Plot dari *Fast frequency hopping* [5]

*Slow frequency hopping* (SFH) terjadi jika salah satu atau lebih simbol informasi yang ditransmisikan dalam interval waktu antara hop frekuensi. *Slow frequency hopping* (SFH) biasanya lebih baik karena gelombang yang ditransmisikan jauh lebih rapat dan *switching time* yang berlebihan bisa berkurang. Pada *Slow frequency hopping* (SFH) ini dimana  $T_c > T_s$ . Sinyal *slow FH/MFSK* adalah karakteristik dengan memiliki beberapa simbol transmisi setiap hop. Oleh karena itu, setiap simbol dari sinyal *slow FH/MFSK* adalah sebuah chip. Ketentuan dari *Slow frequency hopping* (SFH) adalah  $T_c = NT_s$ . Persyaratan orthogonalitas untuk sinyal FSK pemisahan antara simbol frekuensi FSK yang berdekatan setidaknya  $2\pi/T_s$ . Oleh karena itu, pemisahan minimum antara frekuensi hopping berdekatan adalah  $2\pi M/T_s$ . *Slow frequency hopping* (SFH) dengan 4-FSK ( $M=4$ ), 6 frekuensi hop ( $L=6$ ), dan 4 simbol setiap lompatan ( $t_c = 4T_s$ ,  $N=4$ ). [5]



**Gambar 2.4** *Frekuensi Plot dari Slow frequency hopping* [5]

Dimana :  $T_c$  = Hop period

$T_s$  = Symbol period

$T_b$  = Bit period

$W$  = Spread bandwidth

$B$  = Symbol bandwidth

$M$  menunjukkan ukuran *hopset*,  $B$  menunjukkan *bandwidth* kanal frekuensi, dan  $F_s$  menunjukkan pemisah minimum antara frekuensi pembawa yang berdekatan dalam *hopset*. Untuk proteksi terhadap interferensi *narrowband* yang tidak berubah dan *jamming*, diharapkan bahwa  $F_s \geq B$  sehingga spektral kanal frekuensi hampir beririsan.

Kemudian *hop* memungkinkan sinyal yang ditransmisikan dapat terhindar dari gangguan di kanal frekuensi [4].

Untuk mendapatkan keuntungan penuh dari blok atau saluran *convolutional code* dalam sistem *slow frequency hopping* (SFH), penting untuk menyisipkan simbol kode sedemikian rupa bahwa kesalahan simbol dalam *code word* atau *constraint length* dapat berdiri sendiri (untuk *hard-decision decoding*) atau simbol yang didegradasi dengan sendirinya (untuk *soft-decision decoding*). Dalam sistem operasi *frequency-hopping* melalui kanal *frequency-selective fading*, realisasi secara bebas ini memerlukan batasan tertentu antar nilai-nilai parameter sistem. Kesalahan simbol dengan sendirinya jika *fading* secara bebas di setiap kanal frekuensi dan setiap simbol ditransmisikan dalam kanal frekuensi yang berbeda. Jika masing-masing kode simbol disisipkan dan ditransmisikan di lokasi yang sama di setiap *hop dwell interval*, maka simbol yang berdekatan dipisahkan oleh  $T_h$ , setelah proses penyisipan. Dengan demikian, suatu kondisi yang cukup untuk kesalahan simbol hampir bebas adalah

$$T_h \geq T_{coh} \quad (2.1)$$

Dimana  $T_{coh}$  adalah waktu koheren dari kanal *fading*. Kondisi lain untuk kesalahan simbol hampir *independent* adalah

$$F_s \geq B_{coh} \quad (2.2)$$

Dimana  $B_{coh}$  adalah lebar pita koheren dari kanal *fading*. Untuk jaringan komunikasi bergerak dengan *hop rate* melebihi 100 hop/detik, (2.4) jarang terpenuhi. Untuk *hopping band* dengan *bandwidth*  $W$ , dan *hopset* dengan sebuah pemisahan pembawa yang seragam,  $F_s = W/M \geq B$ . Maka, (2.3) menunjukkan bahwa jumlah kanal frekuensi dibatasi oleh [3]

$$M \leq \frac{W}{\max(B, B_{coh})} \quad (2.3)$$

Jika ada kesalahan simbol yang hampir independen harus dipastikan. Jika (2.6) tidak memenuhi, akan ada rugi-rugi karena kesalahan simbol yang berkorelasi. Jika  $B < B_{coh}$ , ekualisasi tidak akan diperlukan karena fungsi transfer kanal hampir datar di setiap kanal frekuensi. Jika  $B \geq B_{coh}$ , salah satu ekualisasi mungkin digunakan untuk mencegah *Intersymbol Interference* (ISI) atau modulasi pembawa jamak (*multicarrier*) dapat dikombinasikan dengan *frequency-hopping* [5].

*Frequency-selective fading* dan efek Doppler menyulitkan untuk mempertahankan fase koheren dari hop ke hop antara frekuensi *synthesizer* di pemancar dan penerima. Selain itu, *time-varying delay*



antara perubahan frekuensi sinyal yang diterima dan dari keluaran *synthesizer* di penerima menyebabkan pergeseran fasa sinyal *dehopped* untuk membedakan setiap interval hop. Dengan demikian, praktis sistem *frequency-hopping* menggunakan non-koheren atau diferensial demodulasi koheren kecuali terdapat sinyal pilot, durasi hop sangat panjang, atau estimasi fase berulang-ulang secara teliti digunakan [5].

Dalam aplikasi militer, kemampuan sistem *frequency-hopping* untuk menghindari interferensi yang berpotensi dinetralkan oleh *repeater jammer* (dikenal sebagai *follower jammer*), yang mana perangkat tersebut melakukan penyadapan sinyal, memprosesnya, dan kemudian mengirimkan *jamming* ke pusat frekuensi yang sama. Agar efektif terhadap sistem *frequency-hopping*, energi *jamming* harus mencapai di penerima sebelum melompat ke kumpulan kanal frekuensi baru. Dengan demikian, *hop rate* adalah faktor yang penting dalam melindungi sistem terhadap *repeater jammer*. Diperlukan *hop rate* dan keterbatasan dari *repeater jamming* dianalisis dalam referensi.

## 2.3 Modulasi MFSK

Pada sistem frekuensi *hopping* ini menggunakan *Multiple Frequency Shift Keying* (MFSK) sebagai modulasi datanya. Frekuensi pembawa yang dipilih dari salah satu frekuensi  $q$  untuk setiap simbol yang ditransmisikan, dan kumpulan dari frekuensi tersebut memungkinkan perubahan frekuensi di setiap *hop*. Sistem FH/MFSK dimana keluaran bit dari generator kode dan input digital digabungkan untuk menentukan frekuensi yang dihasilkan oleh *synthesizer*. MFSK merupakan fase kontinyu dari simbol ke simbol, jika tidak dapat dimodelkan sebagai variabel acak terdistribusi merata secara menyeluruh. Implementasi dari kontinuitas fase sangat dibutuhkan untuk mencegah hamburan spektral yang berlebihan di luar kanal frekuensi [4].

Sistem FH/MFSK, dimana setiap frekuensi  $q$  atau pola di kumpulan MFSK dapat dianggap sebagai frekuensi tengah dari sub kanal MFSK. Oleh karena itu, jumlah efektif kanal frekuensi adalah [4]

$$M_e = qM \quad (2.4)$$

$M$  adalah ukuran *hopset*. Pada implementasi standar, sub kanal  $q$  dari setiap kumpulan MFSK berdekatan, dan setiap kumpulan merupakan kanal frekuensi dalam *hopping band*. Untuk sinyal orthogonal non-koheren, pola MFSK harus benar-benar dipisahkan agar sinyal yang diterima menghasilkan respon yang dapat diabaikan di sub-



kanal yang salah. Seperti yang ditunjukkan selanjutnya, pemisahan frekuensi harus  $f_d = k/T_s$ , di mana  $k$  adalah bilangan bulat *nonzero*, dan  $T_s$  menyatakan durasi simbol. Untuk memaksimalkan ukuran *hopset* pada saat sub-kanal MFSK berdekatan, dipilih nilai  $k = 1$ . Akibatnya, lebar pita kanal frekuensi untuk *slow frequency hopping* dengan banyak simbol per *dwell time* adalah [4]

$$B = \frac{q}{T_s} = \frac{q}{T_b \log_2 q} \quad (2.5)$$

Di mana  $T_b$  adalah durasi bit, dan unsur nilai  $\log_2 q$  untuk peningkatan durasi simbol ketika modulasi *nonbinary* digunakan. Jika *hopping band* memiliki lebar pita  $W$ , maka ukuran *hopset* adalah [5]

$$M = \left\lceil \frac{W}{B} \right\rceil \quad (2.6)$$

## 2.4 Teknik Jamming

Pada teknik *Spread Spectrum* merupakan data tersebar diseluruh spectrum frekuensi membuat sinyal tahan terhadap jamming, maupun salah satu contoh jenis *spread spectrum* yang dibahas dalam Tugas Akhir ini adalah frekuensi hopping (FH). Jamming itu sendiri adalah gangguan yang mengakibatkan kemacetan pada saat penerimaan maupun pengiriman data. Penyebabnya di dalam penerimaan sinyal data biasanya adalah karena interferensi atau gangguan dari sinyal yang mempunyai frekuensi sama atau hampir sama, sehingga akibatnya adalah kesulitan bagi sistem untuk mengetahui data yang hilang. Dalam Tugas Akhir ini akan digunakan dua macam *jamming* yaitu *Single Tone Jamming* (STJ) dan *Multiple Tone Jamming* (MTJ). [6]

### 2.4.1 Single Tone Jamming (STJ)

Sebuah sinyal *jamming* merupakan sebuah pola *Continuous Wave* (CW) yang diletakkan di frekuensi tunggal. Dengan demikian, *Single Tone Jamming* (STJ) merupakan sebuah frekuensi tunggal yang terletak dalam *bandwidth* yang ditentukan dari sinyal yang macet, ditunjukkan pada Gambar 2.5. *Single tone jamming* juga disebut dengan *spot jamming*. [6]



**Gambar 2.5** *Single Tone Jamming* [6]

CW *jamming* untuk sistem FHSS tidak berhasil, akan tetapi berhasil diterapkan untuk target yang sangat sempit seperti telegrafi On-Off Keying (OOK) yang tidak mengubah frekuensi. *Single tone jamming* mungkin berguna terhadap sistem AJ DSSS dengan mengatasi pengolahan *gain* dari sistem tersebut pada penerima dan menyebabkan efek kerusakan pada level *despread*. Ketika daya jammer total tetap, daya lebih dapat ditempatkan dalam satu pola dibandingkan di setiap beberapa pola, sehingga meningkatkan kemungkinan mengatasi pengolahan *gain* [6].

Ketika ada satu pola ditempatkan di kanal data, sedangkan saluran komplementer hanya memiliki *thermal noise*, maka *jammer* dapat meningkatkan kemampuan penerimaan bit data benar, tergantung pada hubungan fase antara pola interferensi, pola data dan magnitudo relatif dari *tones*.

#### 2.4.2 Multiple Tone Jamming (MTJ)

*Multiple Tone Jamming* (MTJ) merupakan *jamming* yang dapat mengganggu sinyal saluran beberapa atau seluruh penerima. *Multiple-tone jamming* mendistribusikan daya *jammer* di antara beberapa pola (*tones*) dan ditunjukkan pada Gambar 2.6.



**Gambar 2.6** *Multiple Tone Jamming* [6]

*Jammer* akan memancarkan lebih dari satu pola (*tone*) yang ditempatkan secara acak atau ditempatkan pada frekuensi tertentu. Jika target khusus sistem komunikasi AJ yang rentan terhadap pola tertentu dan *jammer* mengetahuinya, maka mungkin lebih bijaksana untuk menggunakan pola pada frekuensi tertentu daripada menempatkannya secara acak. *Multitone Jamming* diilustrasikan pada Gambar 2.6. ketika interferensinya adalah pola *jammer* hanya ke kedua kanal data dan kanal komplementer tanpa *noise* [6].

## 2.5 Software Defined Radio (SDR)

*Software Defined Radio* (SDR) merupakan sebagai pemancar atau penerima radio yang menggunakan sebuah teknologi memungkinkan untuk mengubah parameter RF, frekuensi pembawa, jenis modulasi atau daya output menggunakan *software*. Radio digunakan pada peralatan *wireless communication* dimana *software* mengimplementasikan fungsi *physical layer* dan *link layer*. Hal ini sesuai untuk peralatan *single wireless* yang kemudian diprogram ulang menggunakan modulasi yang berbeda, *coding* dan protokol akses. *Software* memiliki fungsi utama untuk dapat diprogram untuk standart yang ada secara fleksibel, dan juga dapat diupdate secara dinamis dengan *software* baru tanpa banyak mengubah *hardware* dan infrastruktur yang sudah ada [7].

Pada perancangan SDR perlu diperhatikan dua hal yang menjadi panduan penting yaitu [7] :

1. Implementasi pada fungsi-fungsi *baseband*, seperti *coding*, modulasi, ekualisasi dan *pulse shaping*.
2. Sistem dapat diprogram ulang untuk menggaransi operasi-operasi pada *multistandard*.

Beberapa keuntungan SDR yang paling signifikan adalah :

1. *Cognitive radio* : SDR dapat memanfaatkan *spectrum* radio yang kurang dimanfaatkan, memiliki potensi untuk meningkatkan jumlah *spectrum* yang sedia.
2. *Interoperability* : SDR dapat berkomunikasi dengan beberapa radio yang tidak kompatibel atau bertindak sebagai jembatan antara radio. SDR ini dapat bertindak sebagai penerjemah untuk semua radio yang berbeda.
3. Mengurangi *Obsolescence* : digunakan untuk mendukung standar komunikasi terbaru.
4. SDR dapat menyesuaikan gelombang untuk memaksimalkan *key metric*. Contoh sebuah gelombang dengan daya rendah dapat dipilih jika kehabisan baterai. Dengan memilih gelombang yang sesuai untuk setiap kondisi, maka radio dapat digunakan dengan optimal.
5. SDR dapat digunakan pada beberapa aplikasi.
6. SDR dapat digunakan untuk pengimplemntasian banyak bentuk gelombang yang berbeda pada analisa *real time*. Sehingga bisa jauh lebih cepat dari pada simulasi.

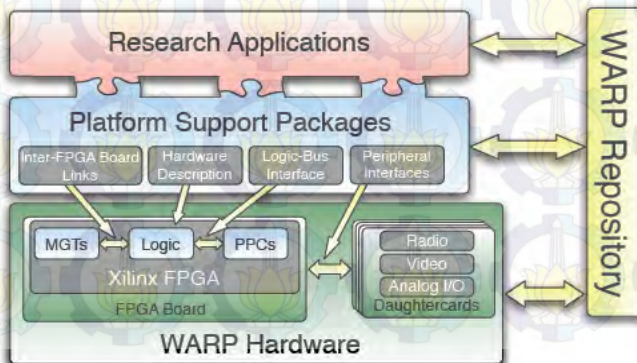


## 2.6 Wireless Open-Access Research Platform (WARP)

*Wireless Open-Access Research Platform (WARP)* merupakan bentuk desain *platform wireless* yang dapat di program untuk membuat *prototype* jaringan *wireless*, sebagai *prototype* komunikasi *wireless* untuk penelitian dalam algoritma nirkabel yang maju saat ini. WARP menggabungkan perangkat dengan kemampuan tinggi yang bisa diprogram dengan *repository open-source*. WARP merupakan salah satu bentuk *Software Defined Radio* yang dikembangkan sebagai modul penelitian tentang sistem komunikasi nirkabel. WARP sebuah *Field Programmable Gate Array (FPGA)* atau modul *Digital Signal Processing (DSP)* yang terhubung dengan modul radio. Pada WARP memiliki Software WARPLab yang berkerja melalui Matlab untuk memberikan perintah pada WARP. Tujuan *platform* dapat digambarkan pada level tinggi karena memiliki empat syarat. Pertama, *platform* mampu mengimplementasikan algoritma nirkabel yang canggih, memungkinkan pengolahan sumber daya tambahan untuk dialokasikan ketika komputasi daya dari proses tunggal membuktikan tidak cukup, memberikan ekspansi *interface* untuk mengakomodasi aplikasi masa depan, dan untuk melestarikan utilitas dalam jangka panjang yang *platform* harus mendukung evolusi beberapa komponen tanpa merusak fungsi lain [8].

### 2.6.1 Arsitektur Platform

*Platform* dirancang untuk memenuhi tuntutan penelitian pada kinerja sistem nirkabel tingkat tinggi. Arsitektur *platform*, digambarkan pada Gambar 2.7 yang terdiri dari empat komponen utama :



**Gambar 2.1** Empat komponen utama dalam WARP [8]

Empat komponen utama dalam WARP adalah sebagai berikut :

1. **Custom Hardware** : dirancang dengan menyesuaikan kebutuhan komunikasi nirkabel dengan kinerja yang tinggi. *Hardware* menyediakan sumber daya tinggi yang mampu diimplementasikan pada kecepatan algoritma, dapat terukur interkoneksinya untuk mengalokasikan lebih banyak kekuatan pemrosesan yang diperlukan, dan dipilih periferil yang diperluas untuk radio, prosesor khusus dan *interface* ekspansi lainnya.
2. **Platform Support Packages** : membutuhkan sejumlah alat dan paket pendukung untuk memungkinkan fungsionalitas dari semua *interface* dan *peripheral*. Paket-paket pendukung mengekspos kemampuan penuh *platform* untuk peneliti yang bekerja pada berbagai *layer*. *Interface* yang baik dirancang untuk membantu menutupi kerumitan pengolahan di *layer* atas dan bawah yang ingin diteliti, yang memungkinkan pengguna untuk mengakses *platform* penuh tanpa harus mengelola kompleksitas penuh.

Paket *platform support* dikembangkan untuk memungkinkan fungsionalitas pada empat tingkat. Tingkat pertama adalah *Low-level Support* yang berfungsi untuk dukungan desain yang diimplementasikan hanya dalam logika. *Logika-PowerPC Support* merupakan tingkatan kedua untuk memungkinkan *interface* antara desain berjalan dalam logika FPGA dan kode yang berjalan di dalam *core PowerPC* (PPC). Prosesor PPC mengandalkan bus standard untuk semua interaksi dengan *programmable logic* sekitarnya. Dengan demikian, desain yang berjalan dalam logika harus mengimplementasikan *interface* ke salah satu bus untuk bertukar data dan informasi control dengan PPC inti. *Peripherals Support* merupakan tingkat ketiga yang bertanggung jawab atas *interface* FPGA dan berbagai off-chip peripheral termasuk dalam *custom hardware*. Lapisan yang termasuk adalah *library* dari modul *interface* yang menjembatani desain yang berjalan di FPGA dengan *card* periferil yang dipasang di *slot board* FPGA *card*. *Board-to-Board Support* merupakan lapisan keempat yang memungkinkan komunikasi *board-to-board* untuk aplikasi yang membutuhkan beberapa FPGA yang bekerja di sistem.

3. **Open-Access Repository** : Repositori berfungsi sebagai arsip untuk semua aspek dari *platform*, termasuk desain, dokumentasi perpustakaan, dan contoh aplikasi. Menerapkan control versi untuk

memungkinkan kolaborasi dengan banyak pengguna pada satu proyek. Isi *repository* tersedia dibawah lisensi terbuka BSD-derived.

4. **Research Applications**: Implementasi algoritma baru, diaktifkan oleh *hardware* dan paket *support platform*. Sistem kendali menggabungkan algoritma baru yang dapat dengan cepat dibangun menggunakan modul blok bangunan standar yang disediakan dalam *repository*.

### 2.6.1.1 Desain Hardware

Komponen utama dari desain *platform* adalah *custom hardware* yang menyediakan proses, *interface* dan *peripheral* perangkat yang diperlukan untuk memenuhi tujuan penjelasan WARP. Revisi awal dari desain *hardware*, yang ditunjukkan pada Gambar 2.8.

#### A. FPGA Board

*Platform hardware* harus menyediakan sumber daya pengolahan yang substansial untuk memenuhi kebutuhan komputasi dari sistem komunikasi nirkabel yang beroperasi pada 100Mb/detik. Algoritma *wireless* memerlukan banyak komputasi *DSP-centric*, namun DSP dan prosesor tidak menyediakan pemrosesan yang cukup untuk kebutuhan itu. WARP memiliki FPGA sebagai prosesor utama karena memberikan pengolahan sumber daya yang luar biasa, blok *programmable logic* didalamnya dapat saling dihubungkan sehingga dapat membentuk unit fungsional yang kompleks. FPGA juga sangat cocok untuk operasi *DSP-intensif*, terutama dalam aplikasi dimana algoritma dapat diparalelkan. Misalnya, pengolahan *front-end* untuk aplikasi nirkabel yang membutuhkan operasi *throughput* yang tinggi seperti filter dan korrelator yang direplikasi untuk setiap *interface* nirkabel. Implementasi dari operasi ini dapat memanfaatkan struktur paralel pada perangkat keras untuk meningkatkan kinerja. Selanjutnya, setiap *instance* dari unit fungsional beroperasi secara paralel dengan semuanya. *Multi-level paralelisme* merupakan cara utama FPGA memberikan kinerja yang jauh di luar kemampuan bahkan untuk DSP yang paling kuat.

FPGA merupakan bagian utama dalam *platform*, sedangkan Xilinx Virtex-II Pro FPGA adalah bagian utama dari desain ini. Selain menyediakan operasi DSP-intensif, FPGA juga menyediakan pilihan interkoneksi yang fleksibel dan cepat untuk antarmuka periferil dan menciptakan sistem multi-prosesor, yang merupakan dua syarat utama dari desain *platform*. Virtex-II Pro juga mencakup core prosesor



PowerPC, menyediakan sumber daya yang ideal untuk menerapkan algoritma lapisan yang lebih tinggi dan cocok untuk prosesor *programmable logic*.

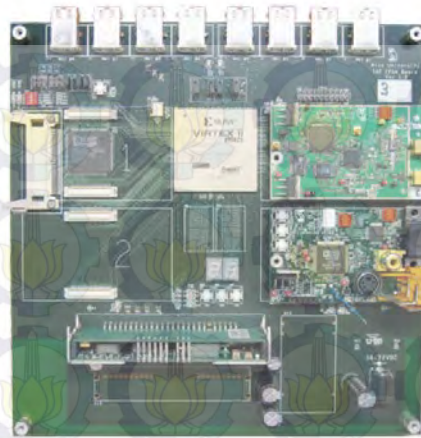
FPGA sendiri menyediakan kekuatan pemrosesan yang besar, koneksi untuk perangkat lain dan board yang memungkinkan berbagai aplikasi. Board FPGA menyediakan antarmuka Ethernet 10/100 untuk koneksi ke jaringan kabel standar. Koneksi ini memungkinkan komunikasi *real-time* antara node jaringan kabel yang ada dengan node jaringan nirkabel kustom yang diimplementasikan pada WARP. Board FPGA dibangun disekitar FPGA t unggal yang besar untuk menghubungkan beberapa FPGA bersama-sama ketika diperlukan. Untuk mengatasi persyaratan skalabilitas dengan menggunakan *multi-gigabit transceiver* (MGT) dibangun ke dalam FPGA Xilinx. Setiap MGT menyediakan *full duplex* 3+ Gb/detik untuk hubungan antara dua FPGA, beberapa MGT dapat digunakan secara paralel untuk menyediakan *throughput* yang lebih antara dua papan. Delapan MGT diarahkan ke konektor *off-board* pada setiap board FPGA, yang menyediakan kemampuan komunikasi substansial (~24 Gb/s) antar FPGA.



**Gambar 2.2** FPGA Board [8]

### **B. Peripheral Daughtercards**

FPGA Board menyediakan empat slot *daughtercard*. Setiap slot terhubung ke pin I/O, menyediakan fleksibel *high-throughput interface*. Slot *daughtercard* juga menyediakan 30 watt gabungan daya yang terisolasi dari pasokan FPGA. Slot *daughtercards*, mulai dari perangkat *interface* seperti radio dan Ethernet untuk modul pengolahan tambahan yang dibangun di sekitar DSP atau prosesor khusus.



**Gambar 2.3** Revisi Pertama dari *Platform Custom Hardware*, termasuk *Board Virtex-II Pro FPGA*, *Radio Daughtercard* dan *Video Daughtercard* [8]

Rancangan dari tiga *daughtercards* pada revisi pertama diperangkat keras. Yang pertama adalah *interface* radio, yang ditunjukkan pada Gambar 2.9, dibangun disekitar chip radio *transceiver direct-conversion* tunggal dari Maxim Integrated Products. Radio Maxim mendukung penuh band ISM/UNII di frekuensi 2,4GHz dan 5GHz, termasuk dukungan untuk saluran dalam band-band yang tidak tersedia di Amerika Serikat tetapi terbuka dinegara lain.

*Daughtercard* kedua menyediakan *video capture* analog dan kemampuan pemutaran. Modul *interface* FPGA untuk *card* ini telah selesai dan menyediakan sarana sederhana bagi desainer algoritma untuk memperoleh, menampilkan gambar dan data video. *Daughtercard* ketiga menyediakan enam saluran analog cepat I/O (2 A/D dan 4 D/A). *Interface* ini memungkinkan palaksanaan algoritma nirkabel secara *real-time* pada frekuensi *baseband*, *decoupling* proses *debugging interface* algoritmik dan RF. Spesifikasi untuk *interface daughtercard* merupakan bagian dari repository *open access*.

### C. *Platform Support Packages*

*Platform* ini membutuhkan sejumlah alat dan paket pendukung untuk memungkinkan fungsionalitas dari semua *interface* dan

peripheral. Paket *platform support* harus dikemangkan untuk memungkinkan fungsionalitas pada empat tingkat. *Low-level Support* merupakan tingkat pertama yang berfungsi untuk mendukung desain yang diimplementasikan hanya dalam logika. Pekerjaan pada tingkat ini hanya membutuhkan Xilinx ISE alat desain untuk mensintesis *hand-coded* HDL dan peta desain untuk perangkat keras. Dukungan *hardware* untuk desain tersebut sebagai besar terdiri dari waktu dan spesifikasi I/O yang khusus untuk *hardware* ini.

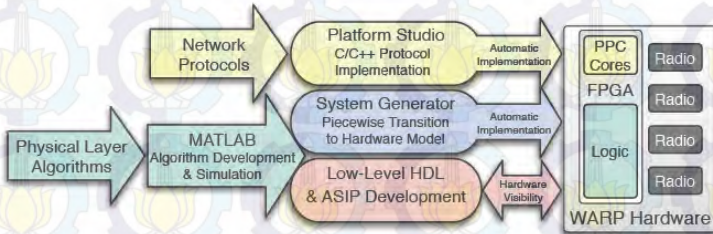
*Logika-PowerPC Support* merupakan tingkat kedua dari paket *support* memungkinkan *interface* antara desain berjalan dalam logika FPGA dan ode yang berjalan di dalam *core PowerPC* (PPC). Prosesor PPC mengandalkan bus standart untuk semua interaksi dengan *programmable logic* sekitarnya, agar desain yang berjalan dalam logika harus mengimplementasikan *interface* ke salah satu bus untuk bertukar data dan informasi control dengan PPC inti. Logika menerapkan alamat *decoding* dan *multiplexing bus* agar *interface* antara bus PPC dan desain sistem generator bekerja dengan baik. Logika ini berhasil digunakan pada *transceiver* OFDM untuk sebagai memori standar yang dipetakan oleh periferil, muncul sebagai *interface* jaringan sederhana untuk program pengendalian.

*Peripherals Support* merupakan tingkat ketiga paket *support* yang bertanggung jawab atas *interface* FPGA dan berbagai *off-chip* peripheral termasuk dalam *custom hardware*. *Library* dari modul *interface* yang menjembatani desain yang berjalan di FPGA dengan *card* peripheral yang dipasang di *slot board* FPGA *card*. Modul *interface* untuk *daughtercard* radio terdapat dalam dua bentuk, pertama adalah *library* sistem generator yang menyediakan *gateway* di lingkungan simulasi yang memetakan secara otomatis untuk konverter analog kartu radio data dan port control. Kedua adalah modul verilog yang berfungsi sama untuk desain tingkat yang lebih rendah diimplementasikan dalam HDL baku.

*Board-to-Board Support* lapisan keempat dari lapisan *support* paket untuk memungkinkan komunikasi *board-to-board* untuk aplikasi yang membutuhkan beberapa FPGA yang bekerja di sistem. Paket *support* menutupi kompleksitas ini dengan membungkus MGT masing-masing dan menghubungkan sebagai *memory-mapped* periferil di bus inti PowerPC. *Custom interface* secara otomatis menangani penyangga yang diperlukan untuk kompensasi jam di kedua pemancar dan



penerima. Perancangan *interface* yang sama untuk digunakan oleh desain tingkat rendah yang tidak memanfaatkan *core PowerPC*.



**Gambar 2.4** WARP Design Flows [8]

### 2.6.1.2 WARP Design Flows

Gambar 2.10 menunjukkan bagaimana para peneliti merancang berbagai lapisan dari jaringan nirkabel *custom* saat menggunakan alat *platform interface* untuk mengintegrasikan implementasi lapisan yang berbeda. Pokok daya dari WARP *design flow*, memberikan contoh bagaimana sebuah jaringan baru seperti spektrum *sniffing* IEEE 802.11 dapat dibangun dengan relative mudah. Menggunakan spektrum *custom* dan manajemen protokol, dikombinasikan dengan standar IEEE 802.11 MAC dan lapisan fisik OFDM dari repository WARP, maka jaringan penuh segera direalisasikan. Jaringan yang dihasilkan dapat secara aktif mencari saluran dengan gangguan sedikit, mudah mengalahkan standar jaringan IEEE 802.11 dengan alokasi kanal statis. Protokol *sniffing* akan diimplementasikan dalam C/CC++ disalah satu prosesor *platform* PowerPC, yang memandang lapisan fisik sebagai jaringan *card interface* menggunakan paket dukungan *platform*. [8]

### 2.6.1.3 Open-Access Repository

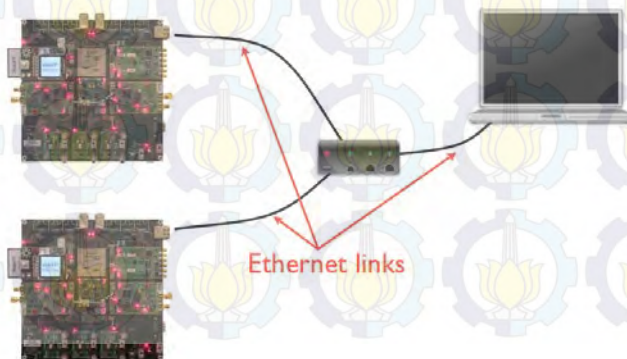
Ketiga komponen utama dari *platform* infrastruktur adalah *open-access repository*. Repositori berfungsi sebagai arsip untuk semua aspek dari *platform*, termasuk desain, dokumentasi, dan contoh aplikasi. Penerapan control versi untuk semua model kode, dan dokumentasi serta menyediakan fitur manajemen proyek dasar untuk memungkinkan kolaborasi dengan banyak pengguna pada satu proyek. Isi repositori tersedia dibawah lisensi terbuka BSD-derived. Isi dari repositori dapat diklasifikasikan ke dalam lima kategori utama :

- Desain *Hardware*: skema, tata letak dan dokumentasi untuk semua papan sirkuit *custom* dicetak.

- b. *Platform Support Packages*: alat dari library memberikan dukungan *interface*.
- c. Aplikasi *Building Blocks*: dioptimalkan dan desain didokumentasi menerapkan berbagai unit fungsional standar. Ini termasuk kedua blok generic, seperti correlators kecepatan tinggi dan *multi-level* filter, serta blok aplikasi spesifik seperti deteksi paket OFDM dan sinkronisasi symbol. Pengguna *platform* dapat memanfaatkan blok *know-good* ini untuk mempercepat desain sistem mereka sendiri.
- d. Penelitian aplikasi: untuk desain *hardware* yang sudah siap. Desain saat ini meliputi implementasi dari lapisan jaringan penuh, seperti PHY OFDM dan protokol MAC sederhana. Kontribusi masa depan akan meliputi MIMO OFDM, LDPC, jaringan coding dan protokol jaringan *multi-hop* yang canggih.
- e. Tutorial, Catatan Aplikasi dan Proyek Mahasiswa: dokumentasi dan contoh-contoh desain yang akan berfungsi sebagai pengantar ke *platform* bagi pengguna baru.

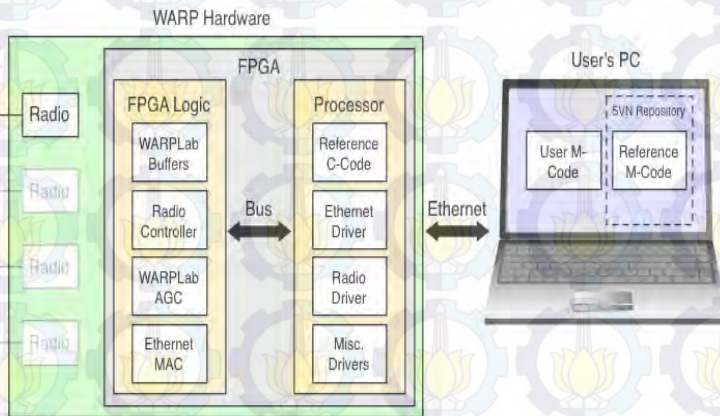
## 2.6.2 WARPLab

WARPLab merupakan kerangka kerja yang digunakan untuk desain penggabungan WARP dan Matlab. Penggunaan dari WARPLab memungkinkan kombinasi dari beberapa sistem multi-antena pada pemancar dan penerima. WARPLab berinteraksi dengan node WARP langsung dari *workspace* Matlab dan sinyal yang dihasilkan dalam Matlab dapat ditransmisikan secara *real-time* melalui udara menggunakan node WARP, untuk gambaran dari WARPLab ditunjukkan pada Gambar 2.11. [9]



**Gambar 2.5** Gambar WARPLab [9]

Masing-masing *node* yang diprogram melalui WARPLab terhubung dengan komputer melalui switch Ethernet. Dengan menggunakan sistem ini, *node* dapat diprogram untuk mengirim data dan menerima data melalui MATLAB. *Setup* WARPLab ditunjukkan pada Gambar 2.12.



**Gambar 2.6** *Setup* WARPLab [10]

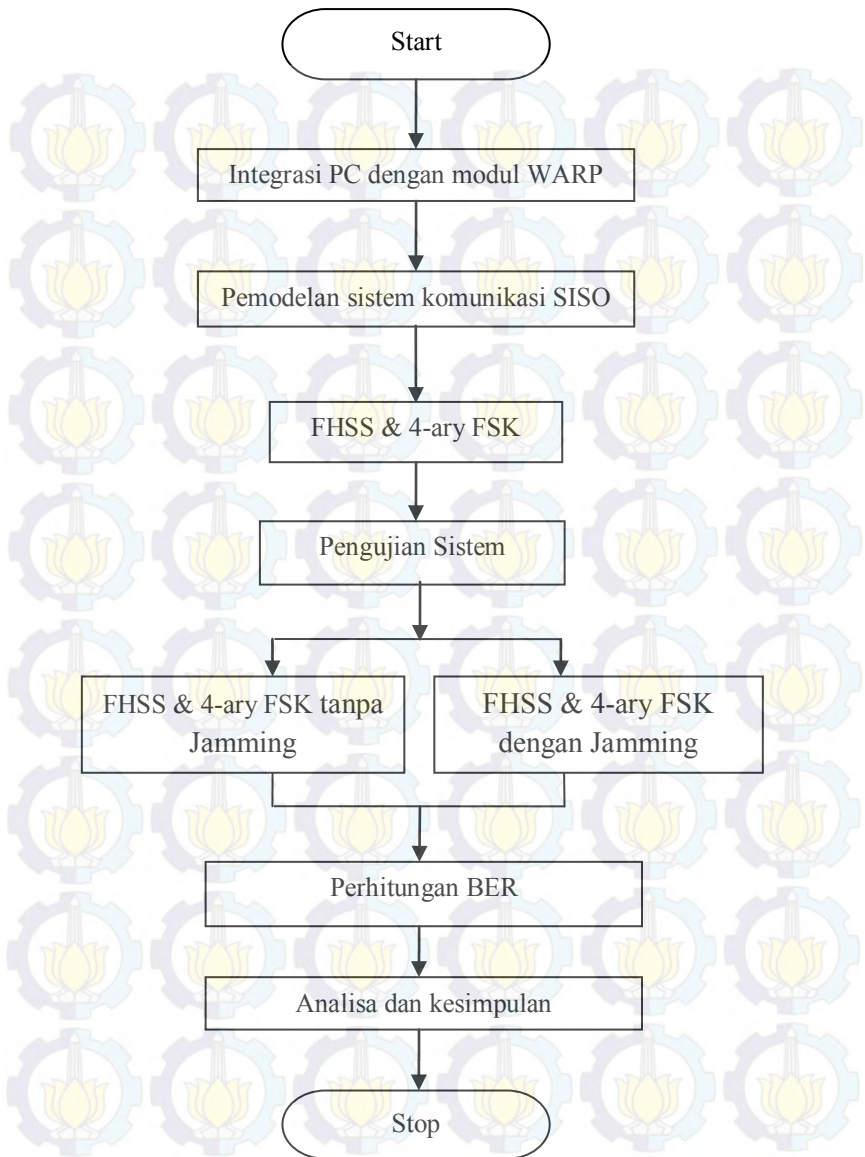


## **BAB III**

### **PERANCANGAN DAN IMPLEMENTASI SISTEM**

#### **3.1 Pendahuluan**

Pada bab ini dibahas tentang proses perancangan dan implementasi sistem *Frequency Hopping Spread Spectrum* dengan menggunakan WARP. Untuk memudahkan proses perancangan implementasi diperlukan langkah-langkah yang harus dilakukan pada saat mengerjakan Tugas Akhir, seperti pada Gambar 3.1. Dalam melakukan perancangan sistem perlu dilakukan beberapa langkah-langkah diantaranya adalah integrasian *Personal Computer* (PC) dengan modul WARP, tahap kedua adalah pemodelan sistem komunikasi SISO dengan *Frequency Hopping Spread Spectrum* (FHSS) penerapan model menggunakan program Matlab untuk mengolah data sebelum dikirim dan setelah diterima secara *offline*. Tahap selanjutnya adalah skenario pengujian sistem yang akan dilakukan di lingkungan *indoor* saja. Dalam tahap pengujian pada sistem *Frequency Hopping Spread Spectrum* menggunakan beberapa jenis *jamming* dan pengujian sistem tanpa *jamming*. Jenis *jamming* yang akan digunakan dalam melakukan pengujian adalah *Single Tone jamming* dan *Multi Tone Jamming*. Selain itu akan dibandingkan dengan sistem 4-ary FSK. Dengan menetapkan skenario pengujian selanjutnya dapat menganalisa serta mengevaluasi sistem yang telah dibuat untuk melihat tingkat keberhasilan sistem tersebut. Sehingga pada akhirnya dapat dilakukan perhitungan serta analisa nilai *Bit Error Rate* (BER) baik menggunakan *jamming* maupun tidak menggunakan *jamming*, begitu juga terhadap perbandingan menggunakan 4-ary FSK. Untuk proses pengukuran implementasi sistem *Frequency Hopping Spread Spectrum* dengan menggunakan WARP dilakukan di lingkungan *indoor* di ruang Laboratorium Komunikasi Multimedia atau ruang B.304.



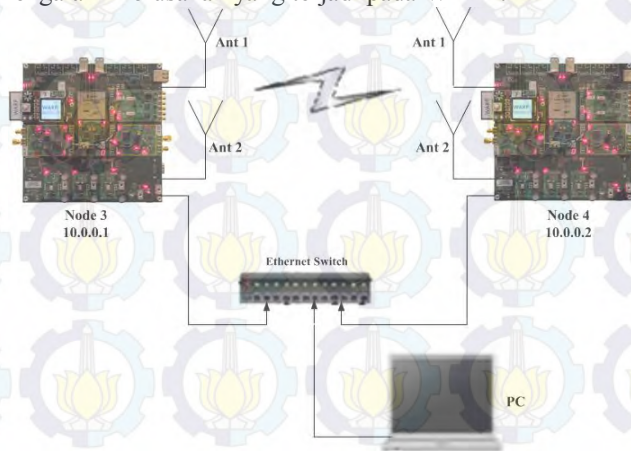
**Gambar 3.1** Blok Diagram Langkah-Langkah Penelitian

### 3.2 Integrasi *Personal Computer* (PC) dengan Modul WARP

Implementasi sistem *Frequency Hopping Spread Spectrum* (FHSS) pada penelitian ini menggunakan modul WARP versi 2. Untuk mengintegrasikan PC dengan WARP menggunakan *Ethernet switch* dan kabel LAN. Penelitian ini menggunakan WARPLab, yang merupakan sebuah *frame work* yang bisa mengintegrasikan modul WARP dengan program Matlab secara langsung. Modul WARPLab berfungsi agar sinyal yang dibangkitkan melalui program Matlab bisa dikirimkan secara *real-time* melalui udara dengan menggunakan *node* WARP pemancar dan penerima menggunakan *switch ethernet* untuk sistem komunikasi langsung dari pemancar ke penerima. Integrasi sistem *Frequency Hopping Spread Spectrum* (FHSS) terhadap PC dengan WARP memiliki dua bagian yaitu integrasi FHSS tanpa *jamming* dan integrasi FHSS dengan *jamming*. Begitu juga terhadap perbandingan menggunakan 4-ary FSK.

#### 3.2.1 Integrasi Sistem FHSS Tanpa *jamming*

Pada sistem komunikasi FHSS tanpa *jamming* ini, data dikirim melalui dua *Node* WARP, seperti Gambar 3.2 menunjukkan *node* 3 (pemancar) ke *node* 4 (penerima) melalui udara, dimana antara node dihubungkan menggunakan *switch ethernet*. Pada Gambar 3.2 antenna 2 tidak berfungsi, karena hanya untuk menutupi pada bagian RF\_B agar tidak mengalami kerusakan yang terjadi pada WARP.

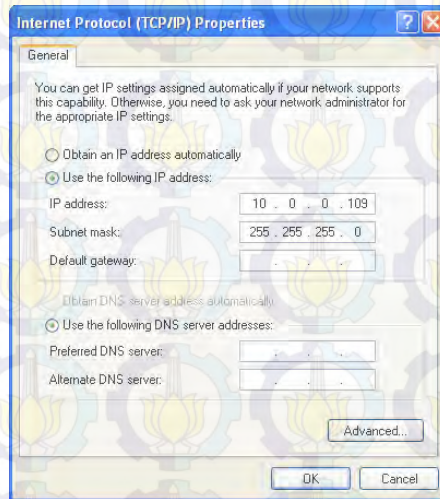


**Gambar 3.2** Konfigurasi Sistem Komunikasi SISO Pada WARP



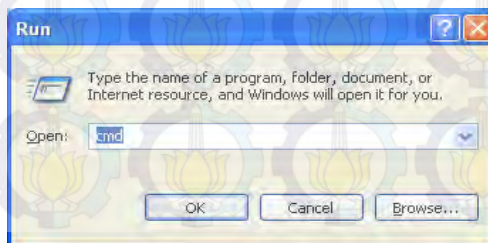
Langkah-langkah untuk mengintegrasikan PC dengan modul WARP versi 2 untuk FHSS tanpa *Jamming*, dan begitu juga untuk 4-ary FSK adalah sebagai berikut :

1. Menghubungkan PC dengan modul WARP dengan *switch ethernet* menggunakan kabel LAN yang ditunjukkan pada Gambar 3.2.
2. Mengatur IP Address LAN pada PC yaitu misalnya 10.0.0.109 dan Subnet mask dengan 255.255.255.0, seperti Gambar 3.3.



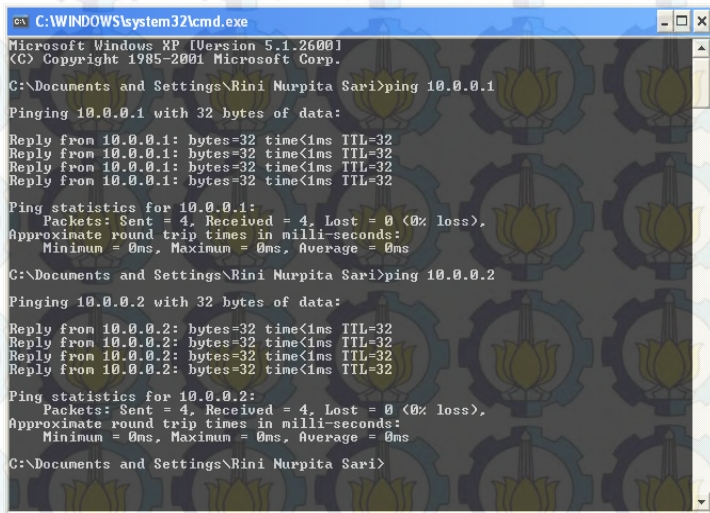
**Gambar 3.3** Pengaturan IP Address FHSS tanpa *Jamming*

3. Memeriksa konfigurasi *Address Resolution Protocol (ARP)* pada *Command Prompt* dengan masuk sebagai administrator.



**Gambar 3.4** Cmd FHSS tanpa *Jamming*

4. Selanjutnya untuk memeriksa *routing Address Resolution Protocol* (ARP) antara WARP dan PC dengan cara melakukan ping terhadap IP *node* 3 (pengirim) dengan IP 10.0.0.1 dan *node* 4 (penerima) dengan IP 10.0.0.2, seperti yang ditunjukkan Gambar 3.5.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Rini Nurpita Sari>ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Documents and Settings\Rini Nurpita Sari>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 10.0.0.2: bytes=32 time<1ms TTL=32
Reply from 10.0.0.2: bytes=32 time<1ms TTL=32
Reply from 10.0.0.2: bytes=32 time<1ms TTL=32
Reply from 10.0.0.2: bytes=32 time<1ms TTL=32

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Documents and Settings\Rini Nurpita Sari>
```

**Gambar 3.5** Kondisi Ping antara *Node* 3 dan *Node* 4

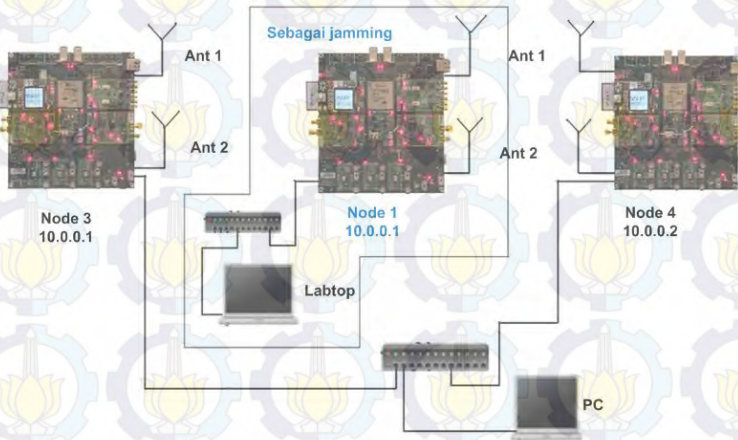
5. Setelah itu memeriksa koneksi antara *Node* WARP dan PC dengan cara pertama buka Matlab, kedua melakukan *setup* WARPLab dengan mengetikkan *wl\_setup* dan tekan tombol “Enter” sampai selesai prosesnya, selanjutnya melakukan cek koneksi dengan cara mengetikkan *wl\_initNodes(2)*. Angka 2 berarti ada 2 buah WARP yang sedang terkoneksi. Untuk indikator berhasilnya koneksi WARP dengan PC dapat dilihat pada output *Command Window Matlab* seperti pada Gambar 3.6.



**Gambar 3.6** Indikator Keberhasilan Koneksi WARP dengan PC

### 3.2.2 Integrasi Sistem FHSS dengan *Jamming*

Pada sistem komunikasi FHSS dengan *jamming*, data dikirim tetap melalui dua *Node* WARP seperti Gambar 3.2, tetapi untuk FHSS yang di *jamming* membutuhkan satu *node* WARP lagi sebagai *jamming*nya. Integrasi FHSS dengan *jamming* tidak jauh berbeda dengan integrasi FHSS tanpa *jamming*, untuk *jamming* butuh *node* 1 yang dijadikan *jamming*. *Node* 1 (*jamming*) akan diintegrasikan dengan PC, seperti yang ditunjukkan Gambar 3.7.

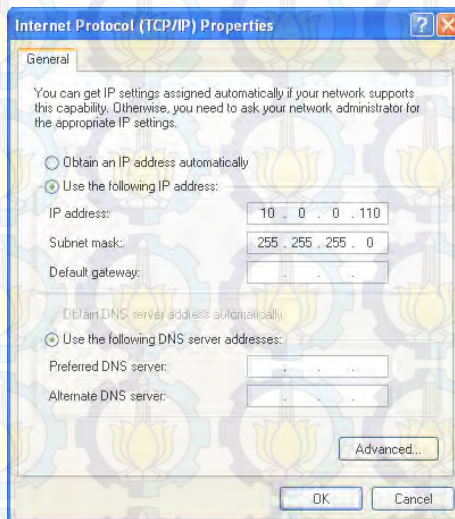


**Gambar 3.7** Konfigurasi Sistem FHSS dengan *Jamming*

Langkah-langkah untuk mengintegrasikan PC dengan modul WARP versi 2 untuk FHSS yang di *Jamming* adalah sebagai berikut :

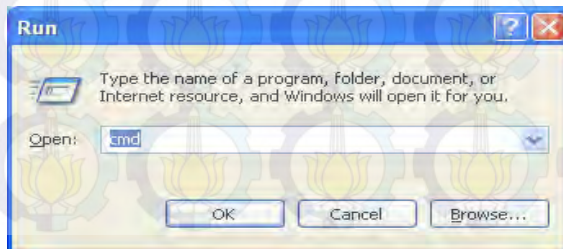


1. Menghubungkan PC dengan *node 1* modul WARP dengan *switch ethernet* menggunakan kabel LAN yang ditunjukkan pada Gambar 3.7.
2. Mengatur IP Address LAN pada PC yaitu misalnya 10.0.0.110 dan Subnet mask dengan 255.255.255.0. untuk IP Address harus beda dengan IP Address FHSS tanpa *jamming*. Seperti yang ditunjukkan Gambar 3.8.



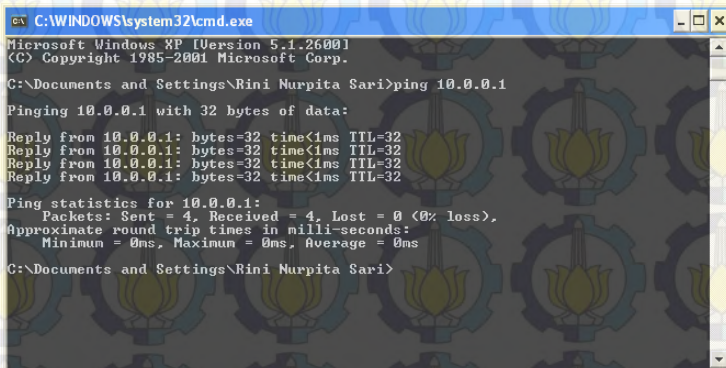
**Gambar 3.8** Pengaturan IP Address FHSS untuk *Jamming*

3. Memeriksa konfigurasi *Address Resolution Protocol* (ARP) pada *Command Prompt* dengan masuk sebagai administrator.



**Gambar 3.9** Cmd untuk *Jamming*

4. Selanjutnya untuk memeriksa *routing Address Resolution Protocol* (ARP) antara WARP dan PC dengan cara melakukan ping terhadap IP *node* 1, seperti yang ditunjukkan Gambar 3.10.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Rini Nurpita Sari>ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Documents and Settings\Rini Nurpita Sari>
```

**Gambar 3.10** Kondisi Ping *Node* untuk *Jamming*

5. Setelah itu memeriksa koneksi antara *Node* WARP dan PC tidak berbeda dengan FHSS tanpa *Jamming* yaitu melakukan *wl\_setup* juga, selanjutnya melakukan cek koneksi dengan cara mengetikkan *wl\_initNodes(1)*. Angka 1 berarti ada 1 buah WARP yang sedang terkoneksi. Untuk indikator berhasilnya koneksi WARP dengan PC dapat dilihat pada output *Command Window Matlab* seperti pada Gambar 3.11.



```
Command Window

Setup Complete
wl_setup():
WARPLOAD v7.1.0
>> wl_initNodes(1)

ans =

Displaying properties of 1 wl_node objects:

  ID | WLVVER | HWVER | Serial # | Ethernet MAC Addr | Address |
-----|-----|-----|-----|-----|-----|
  0 | 7.1.0 | 2 | N/A | 08-50-C2-63-00-00 | 10.0.0.1 |

A >>
```

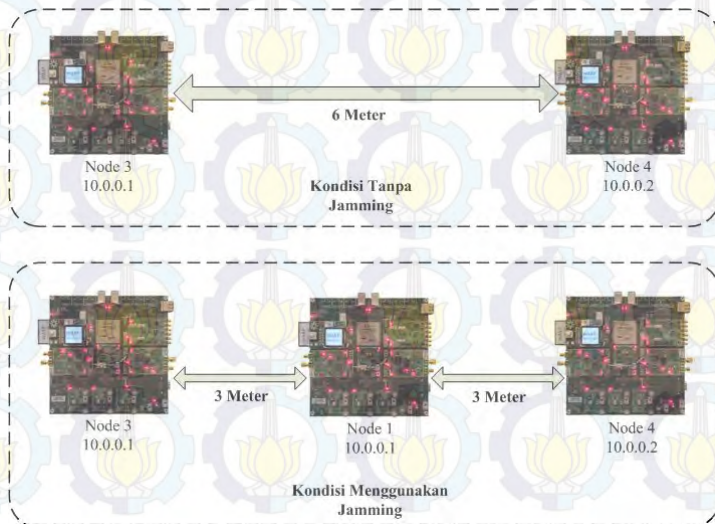
**Gambar 3.11** Indikator Koneksi WARP dengan PC untuk *Jamming*

### 3.3 Skenario Penelitian

Pada penelitian ini menggunakan tiga modul WARP versi 2 sebagai pemancar dan penerima untuk tanpa *jamming*, dan satu lagi sebagai *jamming*. Kondisi untuk melakukan pengukuran dilakukan di ruang B.304 atau Laboratorium Komunikasi Multimedia. Sebelumnya yang sudah dijelaskan pada bab 1 bahwa tujuan dari penelitian ini adalah mengimplementasikan dan evaluasi kinerja *frequency hopping spread spectrum* (FHSS) menggunakan WARP. Parameter yang diteliti dalam Tugas Akhir ini adalah nilai *Bit Error Rate* (BER) fungsi daya pancar dan fungsi jarak dari pengukuran terhadap perbandingan FHSS dan 4-ary FSK tanpa *jamming* dan di *jamming*.

#### 3.3.1 Skenario Pengukuran

Pada sistem pengukuran dilakukan pada jarak 6 meter antara *node* 3 (pemancar) dan *node* 4 (penerima), karena di ruang B.304 Laboratorium Telekomunikasi Multimedia Jurusan Teknik Elektro maksimal sepanjang 6 m eter, sedangkan pengukuran menggunakan *jamming*, *jamming* menggunakan *node* 1 yang di letakkan diantara *node* 3 (pemancar) dan *node* 4 (penerima). Seperti ditunjukkan pada Gambar 3.12.

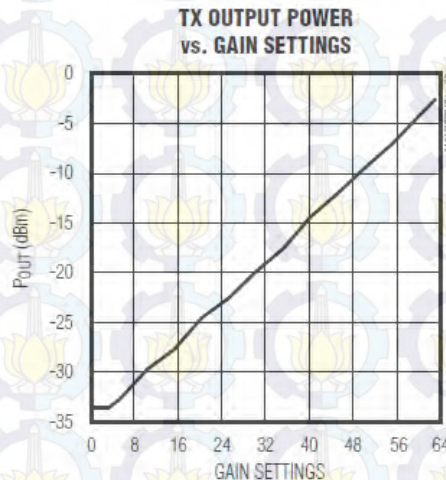


**Gambar 3.12** Skenario Pengukuran



Untuk mengetahui pengaruh nilai daya pancar terhadap BER maka nilai daya pancar diubah-ubah. Pada modul WARP, perubahan nilai daya pancar ditentukan oleh pengaturan gain pemancar pada WARPLab. Secara umum, pengaturan gain dapat dijelaskan sebagai berikut:

a) Transmitter Radio Frequency Gain



**Gambar 3.13** Tx Output Power vs Gain Setting pada MAX2829 [11]

Berdasarkan datasheet MAX2829, karakteristik *baseband* gain memiliki kaitan sebagai berikut :

1. Nilai gain linear pada range 4 hingga 63.
2. Daya output dinyatakan dalam persamaan terhadap gain setting sebagai berikut :

$$P_{out} = 0.5252x - 36.1017 \quad (3.1)$$

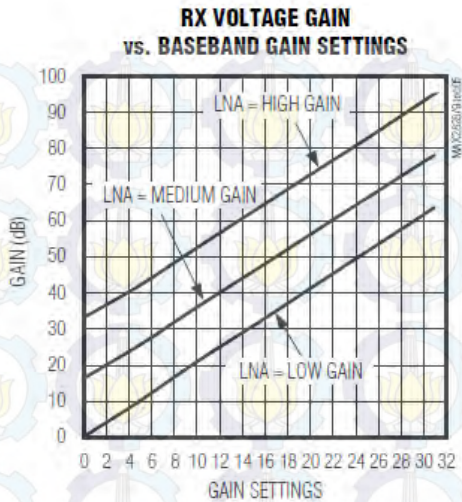
dengan x adalah gain setting dengan range 4 hingga 63.

b) Transmitter baseband gain

Dalam *baseband* gain, terdapat 31 level pengaturan melalui WARPLab.[11]

c) Receiver baseband gain

Gain *baseband receiver*, terdapat 31 level, grafik level pengaturan *baseband* dapat dilihat pada gambar 3.14.



**Gambar 3.14** Gain RX Baseband pada MAX2829 [11]

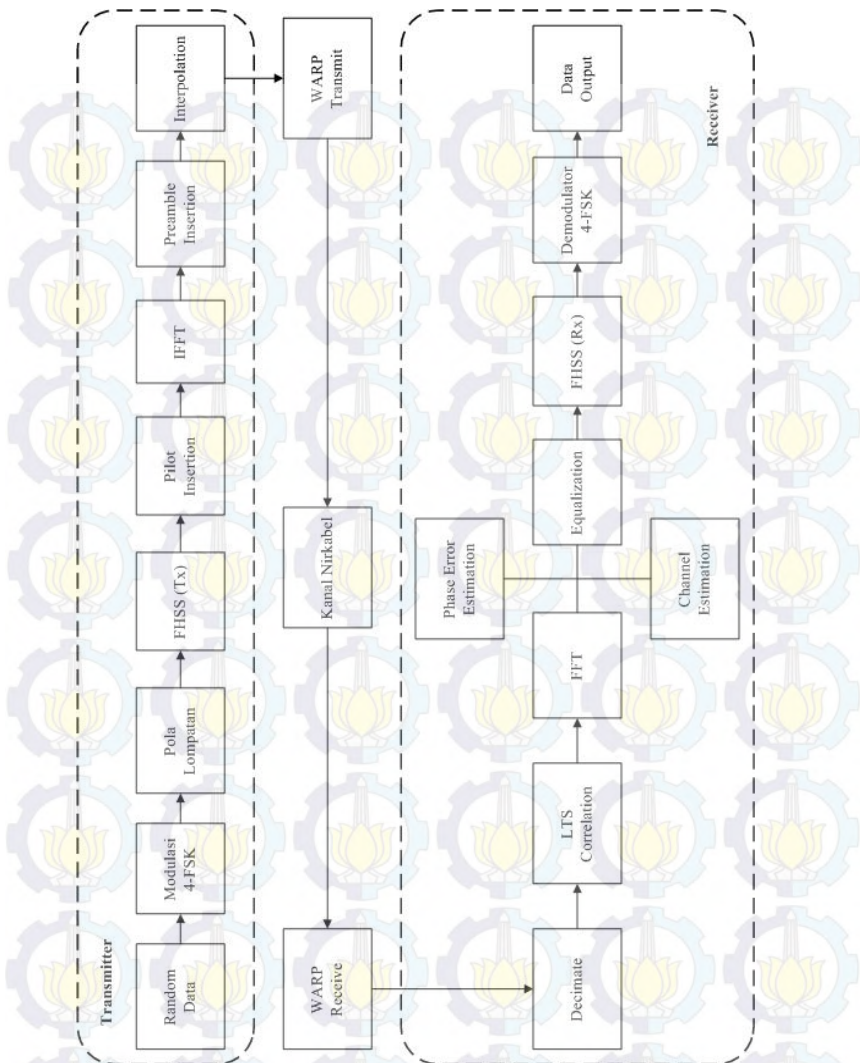
d) Receiver RF gain

Pengaturan RF gain di penerima, *node* WARP memiliki tiga level LNA masing-masing sebagai LNA = Low Gain, Medium Gain dan High Gain. Besar penguatan juga mengacu pada gambar 3.14. Besar penguatan LNA sekitar 0 dB untuk low gain, 15 dB untuk medium gain dan 30 dB untuk high gain.

### 3.4 Desain Sistem Komunikasi FHSS pada WARP

Sebagai acuan dari desain sistem komunikasi *frequency hopping spread spectrum* (FHSS) akan digunakan program yang dibuat oleh Universitas Rice. Pada program ini, dengan menggunakan WARPLab pada Matlab, sistem komunikasi antara dua modul WARP bisa dilakukan. Pada penerapan program Matlab yang sudah di desain akan diterapkan pada WARPLab versi 7. Secara garis besar, proses sistem komunikasi FHSS dilakukan dengan langkah-langkah yang ditunjukkan pada Gambar 3.15.





**Gambar 3.1** Pemodelan Sistem Komunikasi dengan Teknik FHSS

Penelitian pengimplemntasian sistem komunikasi SISO dengan teknik *frequency hopping spread spectrum* (FHSS), menggunakan beberapa parameter yang diatur sebelumnya pada program Matlab untuk

dibutuhkan dalam listing program WARPLab. Parameter-parameter tersebut adalah:

**Tabel 3.1** Parameter Sistem komunikasi SISO dengan Teknik FHSS

No	Parameter	Nilai
1	Modulasi M-ary <i>Frequency Shift Keying</i>	4-ary FSK
2	Jumlah <i>preamble</i>	320
3	Data yang dikirimkan	2160 bit
3	Frekuensi sampling pada modul WARP	40 MHz
4	<i>Carrier channel</i>	5
5	<i>Transmitter RF gain</i>	0:60 [0:63]
6	<i>Transmitter baseband gain</i>	1 [0:3]
7	<i>Receiver baseband gain</i>	1 [0:31]
8	<i>Receiver RF gain</i>	1 [1:3]

### 3.4.1 Pembangkitan Bit Informasi

Pada sistem simulasi FHSS ini, proses pembangkitan bit informasi dibangkitkan secara acak atau *random* dengan menggunakan perintah *randint* pada Matlab dengan *state* tertentu sehingga hasilnya akan saling bebas dan tidak mempengaruhi satu dengan yang lain serta memungkinkan untuk pengulangan atau mendapatkan data yang sama pada *state* yang sama.

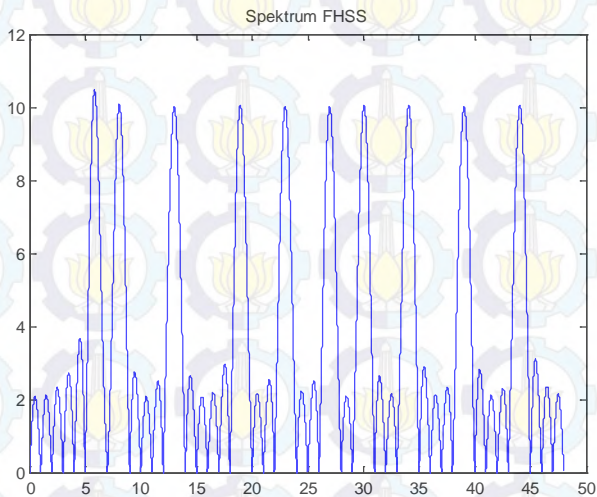
Dalam sistem ini dibangkitkan bit informasi sebanyak 2160 bit secara random, bit yang dibangkitkan mempunyai nilai '0' dan '1' yang merupakan representasi sinyal digital. Bit yang dihasilkan berupa deretan bit secara acak tersebut misalnya (1 1 1 1 1 0 1 1 0 0 1 0 0 0 1 1 1 0 0 1 1 0 1 0 0 1 0 1 1 1 0 1 1 0 1 0 0 1 0 1 0 1 1 0 0 0 0 0 1 0).

### 3.4.2 Proses Modulasi

Bit yang dibangkitkan secara acak berupa bernilai '1' dan '0' ini, selanjutnya bit-bit tersebut sebanyak  $n$  menjadi sinyal yang bernilai *integer* dengan 0 hingga  $(2^n)-1$  sesuai jenis modulasi yang telah ditentukan MFSK ( $m = 2^n$ , dimana  $n = 2, 4, 6$ ). Pada Tugas Akhir ini digunakan  $n = 2$ , sehingga ada 4 kemungkinan nilai simbolnya yaitu [00 01 10 11]. Simbol-simbol tersebut kemudian dikonversi menjadi nilai *integer* (0-3), selanjutnya nilai *integer* tersebut ditambah dengan nilai 1.

### 3.4.3 Pola Lompatan

Pola lompatan yang digunakan dalam proses simulasi sistem *frequency hopping spread spectrum* (FHSS). Nilai pola lompatan adalah nilai acak dari 0 sampai 11, tetapi terus berulang selama sistem berjalan. Pola lompatan yang digunakan pada Tugas Akhir ini adalah '5 9 1 6 2 7 11 4 8 3 10 0', maka pola lompatannya yang akan dihasilkan adalah '5 9 1 6 2 7 11 4 8 3 10 0 ... 5 9 1 6 2 7 11 4 8 3 10 0'. Kemudian pola lompatan ini akan dikalikan terhadap nilai 4-ary FSK dengan slot frekuensi lompatan sesuai dengan pola lompatan yang disimpan pada pola lompatan ini. Setelah proses pola lompatan dibentuk, kemudian akan dibentuk pola FHSS nya dengan matriks 48x120. Gambar 3.16 menunjukkan contoh sampel *Frequency Hopping Spread Spectrum* (FHSS).



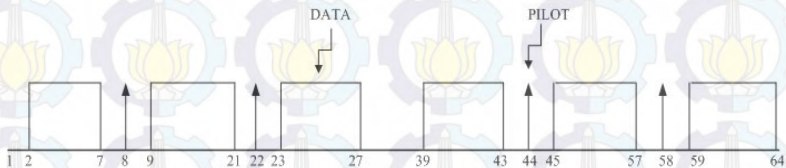
**Gambar 3.2** *Frequency Hopping Spread Spectrum* (FHSS)

### 3.4.4 Inverse fast Fourier transform (IFFT)

Dalam proses IFFT ini memanfaatkan modulasi OFDM untuk pengimplementasian sistem komunikasi SISO pada FHSS dengan menggunakan WARP. Proses IFFT pada Tugas Akhir ini dibentuk dari matriks berukuran 64x120. Kolom 120 menyatakan panjang data yang ditransmisikan. Dalam matriks berukuran 64x120 tersebut terdapat



*subcarrier* pilot yang terletak pada baris (8, 22, 44, 58), sedangkan untuk *subcarrier* data terdapat pada baris (2:7 9:21 23:27 39:43 45:57 59:64). Untuk pola *subcarrier* pilot dan data merupakan referensi dari desain yang digunakan pada Rice University. Urutan antara *subcarrier* data dan *subcarrier* pilot dalam proses IFFT dari Tugas Akhir ini, dapat ditunjukkan pada Gambar3.17.



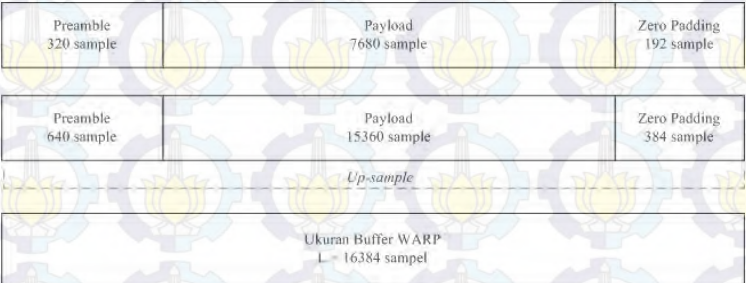
**Gambar 3.3** Ururan *Subcarrier* Data dan *Subcarrier* Pilot

Selanjutnya proses penambahan *preamble*, dimana prosesnya adalah gabungan antara *Long Training Symbol* (LTS) dan *Short Training Symbol* (STS). *Preamble* ini berupa angka scalar. Penelitian yang dilakukan pada FHSS ini menggunakan *preamble* sepanjang 320 bit, urutan *preamble* tersebut (misalnya 0 0 0 0 -1-1i 0 0 0 -1-1i 0 0 0 1+1i). Pada sisi penerima akan menerima bit-bit data yang terdiri dari 320 bit data *preamble* dan deretan bit data yang berupa bilangan kompleks. Kemudian penerima mendeteksi deretan bit *preamble*. Jika sudah terdeteksi maka data setelah *preamble* tersebut dideteksi sebagai data informasi. Standar 320 sampel *preamble* yang akan ditambahkan ke output IFFT. Selain membentuk penambahan *preamble* juga ada pembentukan *payload*, *payload* yang didesain sebesar 7680 sampel dan untuk *zero padding* sebesar 192 sampel. Hasil jumlah nilai *preamble*, *payload*, dan *zero padding* ini belum di *up-sample* ke frekuensi tinggi.

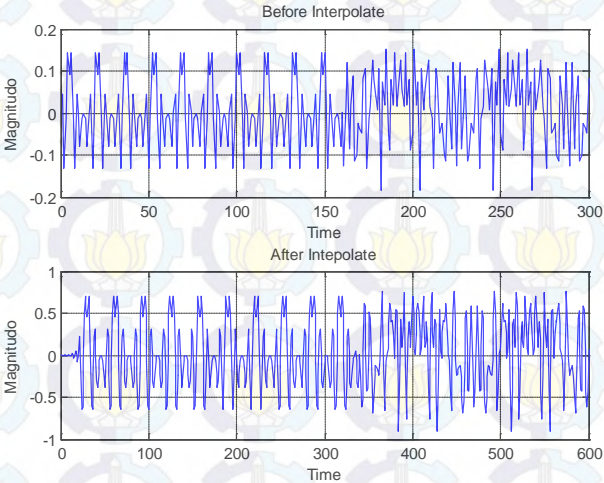
### 3.4.5 Interpolation

Interpolasi merupakan sebuah sampling rate yang lebih tinggi diperlukan untuk mewakili sinyal pada frekuensi yang lebih tinggi. Itu subsistem interpolasi meningkatkan tingkat sampling dari sinyal yang ditransmisikan sehingga dapat kemudian dimodulasikan ke frekuensi yang lebih tinggi dan menjadi sinyal nyata. *Interpolation* diimplementasikan menggunakan kombinasi dari *up-sample* dan filter *low pass filter* (LPF). Pada Tugas Akhir ini, proses interpolasi merupakan proses *up-sample* dua kali dari keluaran nilai pada *zero padding*, dimana

matriksnya  $1 \times 8192$  yang akan disampel menjadi dua kali yang akan menghasilkan matriks sebesar  $1 \times 16384$  ( $2^{14}$ ) untuk memenuhi buffer yang tersedia di WARP sendiri agar bisa dikirim melewati WARP pemancar dan WARP penerima. Proses interpolasi dapat ditunjukkan pada Gambar 3.18. Gambar 3.19 merupakan hasil plot dari blok *interpolation*.



Gambar 3.4 Proses Interpolation



Gambar 3.5 Hasil Plot Blok *Interpolation*

### 3.4.6 Pengiriman Melalui Modul WARP

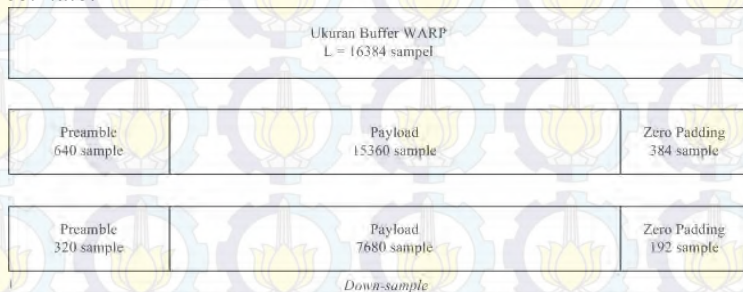
Pada proses melalui transmisi dengan WARP ini, sinyal dikirimkan ke *buffer transmitter* melalui *Ethernet Switch*. Dalam melakukan implementasi ini, untuk pengiriman sinyal melalui modul WARP menggunakan fungsi Matlab yang sudah disediakan oleh WARPLab. Langkah pertama yang harus dilakukan dalam pengiriman melalui WARP adalah menginisialisasi soket-soket yang digunakan. Setelah itu WARP mendownload parameter-parameter yang dibutuhkan. Parameter-parameter tersebut adalah *Carrier channel*, *Transmitter baseband gain*, *transmitter RF gain*, *Receiver baseband gain*, *Receiver RF gain*, yang nilainya terdapat pada Tabel.3.1.

### 3.4.7 Proses di Penerima

Pada penerima terdapat beerapa proses yang harus dilakukan yaitu *decimate*, *LTS Correlation*, *FFT*, *Phase Error Estimation*, *Channel Estimation*, *Equalization*, dan *Demodulator 4-ary FSK*.

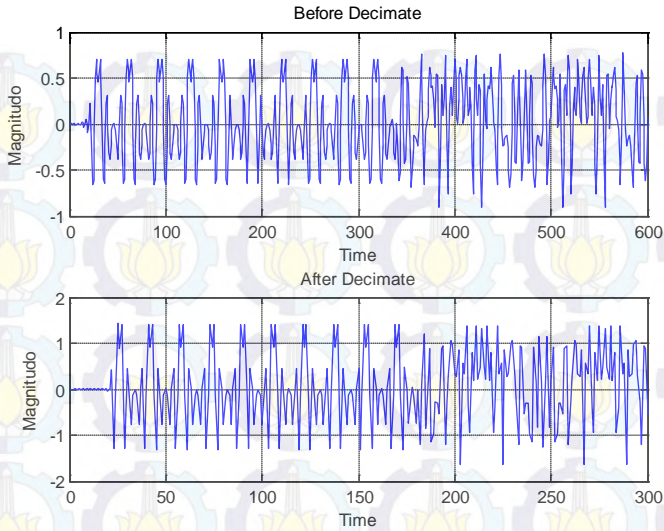
#### 3.4.7.1 Decimate

*Decimate* merupakan proses untuk mengurangi jumlah data yang mewakili sinyal untuk meningkatkan kinerja pemrosesan sinyal. Informasi dalam satu bingkai perlu masuk dalam 64 titik FFT tunggal. *Decimate* ini pada implementasi ini merupakan kebalikan dari proses interpolasi yaitu menurunkan sample rate. Dalam proses *decimate* menggunakan dua tahap proses yaitu fiterl *low pass filter* (LPF) dan *down-sample*. Pada proses *decimate* ini nilai matriks *interpolate* 1x16384 akan menjadi 1x8192, proses *decimate* dapat ditunjukan pada Gambar 3.20, dan Gambar 3.21 merupakan hasil keluaran dari blok *Decimate*.



**Gambar 3.6** Proses *Decimate*





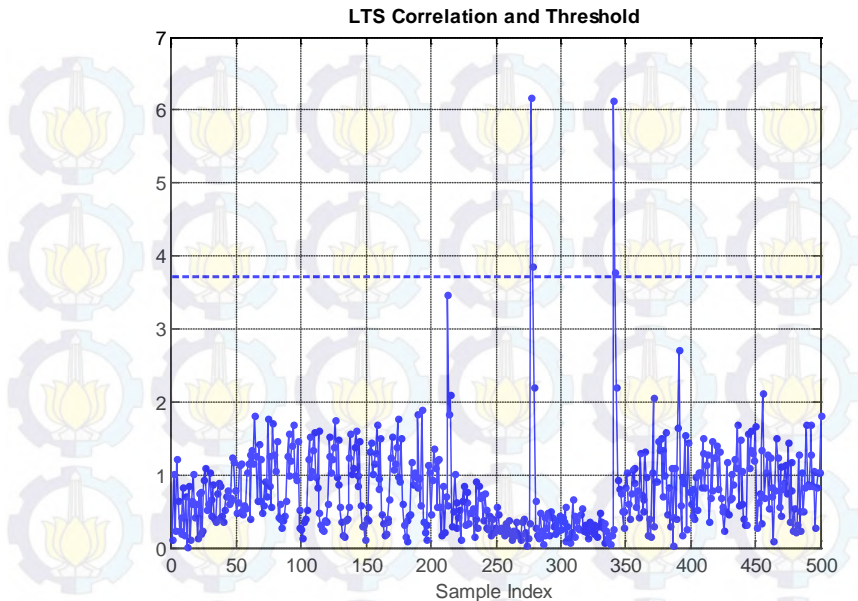
**Gambar 3.7** Hasil Plot Blok *Decimate*

#### 3.4.7.2 LTS *Correlation*

Proses selanjutnya adalah *LTS Correlation* yang merupakan untuk proses sinkronisasi. Pada *LTS Correlation* ini, merupakan proses *cross correlation* antara *preamble* yang terdapat pada penerima dan satu LTS yang terdapat pada pemancar. Proses *cross correlation* merupakan proses untuk melakukan estimasi kanal dan menentukan awal dari *frame* FHSS yang pertama. Pada Tugas Akhir ini, *cross correlation* menggunakan persamaan rumus sebagai berikut [13] :

$$C(n) = \sum_{l=0}^M \sum_{k=1}^N r(l*N + k + n)s^*(l*N + k) \quad (3.2)$$

Dimana,  $r$  merupakan seluruh *preamble*,  $s$  adalah *cross correlation* dengan satu symbol pendek,  $N$  adalah panjang data FHSS dan  $M$  adalah jumlah dari LTS di *cross correlation*. Gambar *LTS Correlation* dapat ditunjukkan pada Gambar 3.22.



**Gambar 3.8** *LTS Correlation*

#### **3.4.7.3 Fast Fourier Transform (FFT)**

Selanjutnya proses FFT dengan ukuran matriks yang dihasilkan FFT harus sama dengan IFFT yaitu  $64 \times 120$ , karena FFT merupakan kebalikan dari proses IFFT. FFT merupakan osilator local pada penerima yang memisahkan frekuensi *carrier* dengan simbol-simbol FHSS yang berbeda pada frekuensi tersebut.

#### **3.4.7.4 Equalization**

Setelah proses FFT langkah selanjutnya yaitu proses equalization, merupakan proses untuk mengatasi nilai magnitudo dan respon fasa yang rusak. Mengatasi nilai magnitudo menggunakan hasil dari *cross-correlation* sedangkan untuk mengatasi respon fasa menggunakan pilot. Proses equalizer terdapat dua proses yaitu pertama proses estimasi kanal untuk domain frekuensi LTS dan kedua proses *phase error estimation*.

Estimasi kanal berfungsi untuk memberikan kompensasi untuk distorsi fasa dan distorsi amplitudo pada sinyal yang terkena dampak

kanal *multipath*. Dalam proses estimasi kanal sebuah penerima *long traning symbol* (LTS)  $R_{l,k}$  ( $l = 0, 1$ ) telah melalui blok FFT dinyatakan dalam bentuk dimana sebuah *noise*  $W_{l,k}$  adalah ditambah ke sebuah hasil dari pemancar *long training symbol* (LTS)  $d_{l,k}$  dan anal  $H_{l,k}$  [14].

$$R_{l,k} = H_{l,k} d_{l,k} + W_{l,k} \quad (3.3)$$

Estimasi kanal menggunakan *long training symbol* (LTS) adalah sebagai berikut [14] :

$$\begin{aligned} \widehat{H}_k &= \frac{1}{2} (R_{0,k} + R_{1,k}) \frac{1}{d_k} \\ &= \frac{1}{2} (H_k d_k + W_{0,k} + H_k d_k + W_{1,k}) \frac{1}{d_k} \\ &= H_k + \frac{1}{d_k} \frac{(W_{0,k} + W_{1,k})}{2} \end{aligned} \quad (3.4)$$

Setelah itu dilakukan proses ekualisasi terhadap *payload* yang diterima, berdasarkan persamaan berikut [15] :

$$xe(k) = x(k) / \widehat{H}_k \quad (3.5)$$

Dimana  $xe(k)$  merupakan sinyal setelah melalui proses ekualisasi,  $x(k)$  yaitu sinyal sebelum melalui proses ekualisasi dan  $\widehat{H}_k$  nilai estimasi kanal.

Proses *phase error estimation* untuk sisa dari domain frekuensi pilot tones. Proses estimasi fasa error ini yaitu dengan mengalikan symbol yang diterima dengan  $e^{j\theta}$  dimana  $\theta$  adalah sudut dari estimasi fasa. Operasi perbaikan fasa dilakukan dengan persamaan sebagai berikut :

$$\dot{s}(k) = s(k)e^{-j\theta} \quad (3.6)$$

$\dot{s}(k)$  merupakan nilai sinyal diterima setelah proses korelasi fasa,  $s(k)$  adalah sinyal diterima sebelum proses koreksi fasa, sedangkan  $\theta$  adalah



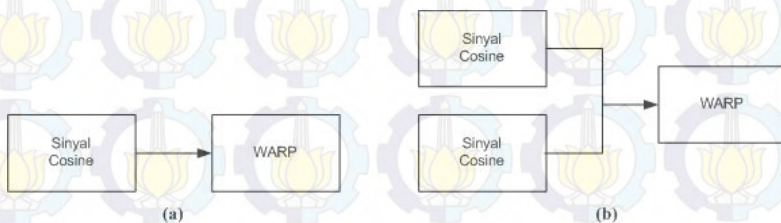
sudut dari estimasi fasa. Nilai  $\theta$  didapat dari rata-rata perbedaan nilai sudut antara pilot yang diterima dengan pilot yang dipancarkan.

#### 3.4.7.5 Demodulator 4-ary FSK

Selanjutnya masuk ke proses demodulasi yang merupakan kebalikan dari fungsi modulasi, proses demodulasi selesai maka didapat keluaran bit informasi yang memiliki jumlah yang sama antara bit informasi yang di bangkitkan pada pemancar.

### 3.5 Desain Sistem Komunikasi FHSS dengan *Jamming* pada WARP

Pada pengujian dengan *jamming* dalam Tugas Akhir ini menggunakan jenis *jamming* yaitu *Single Tone Jamming* (STJ) dan *Multi Tone Jamming* (MTJ). Dalam melakukan pengujian menggunakan *jamming* ini, sinyal daya *jamming* yang akan dirubah-rubah dari 10%, 30%, 50%, 75%, 100%, 125%, 150%, 200%, 225%, 250%, 275%, 300%. Sketsa pengujian menggunakan *jamming* dapat ditunjukkan pada Gambar 3.7. Gambar 3.23 menggambarkan tentang blok diagram sinyal *jamming* yang digunakan.



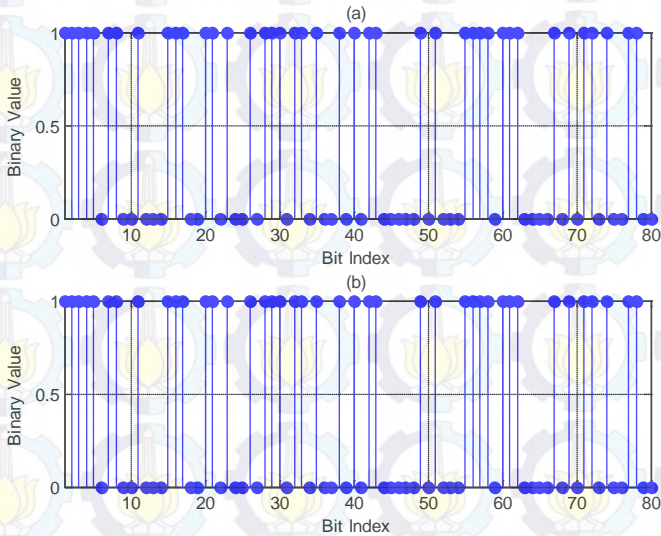
**Gambar 3.9** Blok Diagram *Jamming* (a) *Single Tone* (b) *Multi Tone*

### 3.6 Perhitungan BER

Perhitungan BER pada penelitian ini, data berupa sinyal digital yang dibangkitkan dari dalam Matlab akan diubah kembali ke dalam deretan bit yang terdiri atas satu baris. BER dihitung dengan membandingkan antara deretan bit pada pengirim dengan deretan bit yang dideteksi pada sisi penerima kemudian jumlah bit yang salah

dibagi dengan jumlah bit yang dikirimkan. Adapun hasil dari bit yang dikirim dan diterima dapat dilihat pada Gambar 3.24.

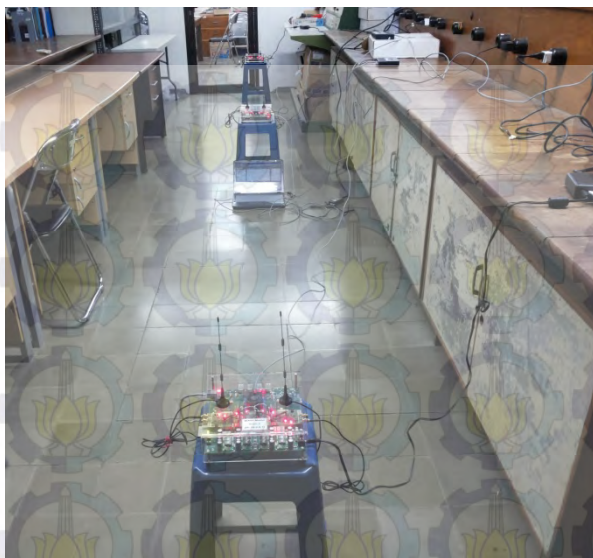
$$BER = \frac{\text{Jumlah bit yang salah}}{\text{Jumlah bit yang dikirim}} \quad (3.7)$$



**Gambar 3.10** (a) Bit yang Dikirim (b) Bit yang Diterima

### 3.7 Metode Pengukuran

Pengukuran berikut dilakukan untuk mengetahui kinerja sistem komunikasi SISO bahwa mampu mengirimkan bit dalam kondisi ideal. Dalam penelitian Tugas Akhir ini dilakukan pengukuran di lingkungan *indoor* yang dilakukan di ruangan B.304 Laboratorium Telekomunikasi Multimedia Jurusan Teknik Elektro. Pada penelitian ini proses pengukuran dilakukan di lingkungan *indoor* pada jarak 6 meter, baik untuk proses pengukuran menggunakan *jamming* maupun tanpa menggunakan *jamming*. Metode pengukuran dapat ditunjukkan pada Gambar 3.25 dan Gambar 3.26.



**Gambar 3.11** Pengukuran Tanpa *Jamming*



**Gambar 3. 12** Pengukuran Menggunakan *Jamming*



## **BAB IV**

### **PENGUJIAN DAN ANALISIS DATA**

Dalam bab ini akan dibahas mengenai pengujian implementasi sistem komunikasi SISO dengan teknik *Frequency Hopping Spread Spectrum* (FHSS) menggunakan *Wireless Open Access Research Platform* (WARP). Hasil pengujian kinerja sistem FHSS secara implementasi akan disajikan dalam bentuk grafik BER vs Daya Pancar (dBm), sedangkan untuk kinerja sistem FHSS terhadap *Single Tone Jamming* dan *Multi Tone Jamming* dalam bentuk grafik BER vs Daya Jamming (%). Terdapat beberapa model sistem FHSS yang dapat dianalisa dari Tugas Akhir ini yaitu Kinerja sistem FHSS tanpa jamming, Kinerja sistem FHSS terhadap gangguan *Single Tone Jamming* (STJ), dan Kinerja sistem FHSS terhadap gangguan *Multi Tone Jamming* (MTJ). Selain dilakukan pengujian terhadap tiga model sistem FHSS diatas, juga akan dilakukan pengujian sistem 4-ary FSK sebagai pembanding. Pada implementasi sistem *Frequency Hopping Spread Spectrum* (FHSS) ini, dilakukan 7 kali pengujian. Kondisi lingkungan pengukuran dilakukan di ruang B.304 Laboratorium Telekomunikasi Multimedia Jurusan Teknik Elektro. Pengujian dilakukan di ruang B.304 dengan jarak 6 meter. Jarak antara *node* pengirim (WARP node 3) dan penerima (WARP node 4) berjarak sejauh 6 meter.

#### **4.1 Analisis Kinerja Implementasi Sistem tanpa Jamming**

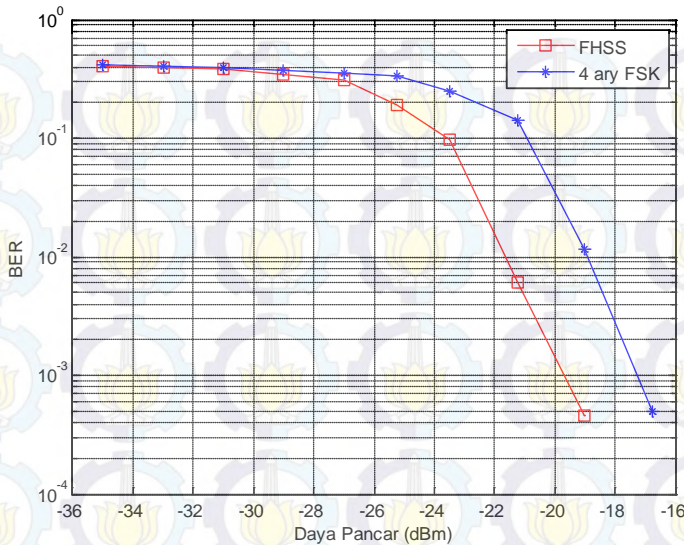
Pengukuran sistem *Frequency Hopping Spread Spectrum* (FHSS) pada implementasi modul WARP menggunakan parameter pengujian dengan nilai Tx RF = 0:4:60, Tx BB = 1, Rx RF = 1, dan Rx BB = 1. Untuk hasil pengujian pada modul WARP ini pada sistem FHSS akan disajikan dalam bentuk grafik BER vs Daya Pancar (dBm), dan dibandingkan dengan 4-ary FSK. Hasil pengujian tanpa *jamming* dapat dilihat pada Tabel 4.1 dan Gambar 4.1.

Berdasarkan Tabel 4.1 dan Gambar 4.1 dapat dilihat bahwa pada implementasi dengan menggunakan modul WARP, kinerja sistem FHSS lebih baik dari pada sistem 4-ary FSK. Pada saat Tx Rf bernilai 0 nilai BER yang dihasilkan sistem FHSS adalah 0.4032, sedangkan untuk sistem 4-ary FSK nilai BER yang dihasilkan 0.4156, dari nilai BER yang dihasilkan pada saat Tx Rf bernilai 0 sudah terlihat perbedaan nilai

BER jauh berbeda. Pada saat Tx Rf bernilai 0 ini, untuk nilai daya pancar yang dihasilkan sebesar -35 dBm. Dengan daya pancar sebesar -35 dBm ini masih sangat kecil maka rentan terhadap *noise* yang mempengaruhi pengiriman informasi, sehingga mengakibatkan banyak informasi yang diterima salah dipenerima. Nilai BER yang paling kecil adalah pada sistem FHSS dari pada sistem 4-ary FSK, berarti bahwa sistem FHSS lebih tahan terhadap *noise* yang akan mempengaruhi informasi ke penerima.

**Tabel 4.1** Hasil Pengujian FHSS dan 4-ary FSK tanpa *jamming*

Tx Rf	Bit Error Rate	
	FHSS	4-ary FSK
0	0.4032	0.4156
4	0.3995	0.4022
8	0.3875	0.3955
12	0.3444	0.3778
16	0.3060	0.3522
20	0.1933	0.3344
24	0.0966	0.2503
28	0.0061	0.1407
32	0.00046296	0.0117
36	0	0.0005
40	0	0
44	0	0
48	0	0
52	0	0
56	0	0
60	0	0



**Gambar 4.1** Grafik BER vs Daya Pancar (dBm) untuk Sistem FHSS dan 4-ary FSK tanpa *Jamming*

Nilai Tx Rf diubah menjadi 4, menghasilkan nilai BER untuk FHSS sebesar 0.3995, sedangkan untuk 4-ary FSK menghasilkan nilai BER sebesar 0.4022. Pada Tx Rf ini, karena daya yang dipancarkan lebih besar dari Gain Tx Rf 0 yaitu bernilai -33 dBm. Dari perbandingan nilai Tx Rf saat 4 dan Tx Rf saat 0 ini, menghasilkan nilai daya pancar yang semakin besar untuk Tx Rf saat 4, sehingga menghasilkan nilai BER yang semakin kecil. Untuk nilai BER yang dihasilkan semakin kecil maka ketahanan terhadap *noise* akan semakin besar pula.

Jika Gain Tx Rf di naikan kembali yaitu sebesar 8 menunjukkan bahwa daya yang dipancarkan sebesar -31 dBm. Nilai BER untuk FHSS adalah 0.3875 dan 4-ary FSK sebesar 0.3955. Nilai BER untuk semua level FHSS dan 4-ary FSK semakin kecil jika dibandingkan saat Tx Rf saat 4, karena daya yang dipancarkan semakin besar maka akan lebih tahan terhadap *noise* dan informasi yang dikirim akan semakin sedikit terjadi kesalahan. Untuk nilai Gain Tx Rf dinaikan kembali menjadi 12, menghasilkan nilai BER FHSS sebesar 0.3444 dan 4-ary FSK sebesar 0.3778 dengan daya yang dipancarkan sebesar -29 dBm. Saat Tx Rf



bernilai 12 ini menghasilkan nilai daya pancar yang lebih besar dari Tx Rf bernilai 8, sehingga mengakibatkan nilai BER semakin kecil untuk FHSS dan 4-ary FSK yang diartikan bahwa semakin kecil terjadi kesalahan informasi yang dikirim.

Nilai Tx Rf di ubah menjadi 16 menghasilkan nilai daya pancar sebesar -27 dBm, dengan begitu nilai BER untuk FHSS adalah 0.3060 dan 4-ary FSK sebesar 0.3522. Pada saat Tx Rf sebesar 16 ini, nilai BER yang dihasilkan lebih kecil dari pada Tx Rf bernilai 12. Kemudian, untuk nilai Tx Rf dinaikan kembali menjadi 20 maka menghasilkan daya pancar lebih besar dari pada Tx Rf adalah sebesar -25.25 dBm, dengan begitu menghasilkan nilai BER antara FHSS dan 4-ary FSK lebih kecil dari pada nilai BER Tx Rf saat bernilai 16. Setelah itu nilai Tx Rf dinaikan kembali menjadi 24, dengan menghasilkan nilai daya pancar sebesar -23.5 dBm. Berdasarkan nilai daya pancar yang dihasilkan semakin besar dari pada daya pancar Tx Rf 20, maka nilai BER yang dihasilkan semakin menurun juga untuk nilai BER saat Tx Rf 24 pada sistem FHSS sebesar 0.0966 dan 4-ary FSK sebesar 0.2503. Dengan arti bahwa data informasi yang dikirimkan lebih sedikit mengalami kesalahan, karena nilai BER yang dihasilkan semakin menurun atau semakin kecil.

Tx Rf dinaikan menjadi 28 akan menghasilkan nilai BER akan semakin kecil, karena nilai daya pancar yang dihasilkan semakin besar yaitu sebesar -21.25 dBm. Setelah nilai Tx Rf bernilai 28 di naikan kembali menjadi 32 akan menghasilkan nilai BER yang jauh lebih kecil dari nilai Tx Rf pada saat 28. Saat Tx Rf bernilai 32 ini menghasilkan nilai BER untuk sistem FHSS sebesar 0.00046296 dan 4-ary FSK adalah 0.0117, karena nilai daya pancar yang dihasilkan pada saat Tx Rf 32 semakin besar yaitu -19 dBm. Sehingga dengan semakin besar daya yang dipancarkan, mengakibatkan data informasi yang dikirim mengalami sedikit kesalahan.

Pada Tx Rf 36 menghasilkan nilai BER untuk sistem FHSS sudah mencapai nilai BER 0, sedangkan untuk 4-ary FSK masih menghasilkan nilai BER sebesar 0.0005 dengan daya yang dipancarkan juga semakin besar yaitu -16.75 dBm. Nilai BER sistem FHSS sudah mencapai 0 ini menunjukkan bahwa daya pancar sudah cukup untuk mengirimkan informasi berupa bit sebanyak 2160 bit. Hal ini menunjukkan bahwa data informasi yang dikirim jauh lebih sedikit mengalami kesalahan untuk dikirim dari pada Tx RF sebelumnya.

Pada saat Tx Rf sebesar 40 nilai BER sistem FHSS tetap 0 dan 4-ary FSK juga sudah mencapai nilai BER 0. Pada Tx Rf sebesar 40 dengan daya pancar -14.5 dBm, sistem FHSS dan 4-ary FSK sudah sama-sama mencapai nilai BER 0, karena daya yang dipancarkan sudah cukup untuk mengirimkan informasi dan *noise* yang ada tidak dapat mempengaruhinya. Grafik BER fungsi daya pancar terhadap pengukuran hasil tanpa *jamming* menunjukkan bahwa semakin besar daya yang dipancarkan maka semakin kecil nilai BER yang dihasilkan dengan jarak antara *node* 3 (pemancar) dan *node* 4 (penerima) tetap. Hal ini dipengaruhi oleh daya yang dipancarkan pada sisi pengirim, semakin besar daya yang dipancarkan maka pengaruh *noise* yang ada untuk mempengaruhi informasi dalam pengiriman data juga akan semakin kecil, serta daya yang diterima di penerima semakin besar pula, sehingga mengakibatkan nilai BER yang dihasilkan pada sistem FHSS dan 4-ary FSK akan semakin kecil. Jadi dapat disimpulkan dari hasil pengukuran implementasi pada modul WARP terhadap sistem FHSS dan 4-ary FSK, dimana nilai BER berbanding terbalik dengan daya pancar.

## 4.2 Analisis Kinerja Implementasi FHSS dengan Gangguan *Jamming*

Pengujian sistem dengan menggunakan modul WARP. Dalam pengujian sistem ini akan diuji dengan dua model gangguan *jamming*, yaitu *Single Tone Jamming* (STJ) dan *Multi Tone Jamming* pada nilai daya pancar (Ptx) bernilai -19 dBm dan -23 dBm. Pada implementasi ini, nilai BER yang didapat dari masing-masing model akan dibandingkan terhadap perubahan daya sinyal *jamming*. Setiap modelnya akan dipengaruhi sinyal *jamming* dengan frekuensi berbeda agar terlihat pengaruh sistem yang didesain. Pada pengujian sistem dengan *Single Tone Jamming* dan *Multi Tone Jamming*, parameter yang digunakan adalah perubahan frekuensi sinyal *jamming* dan daya *jamming* terhadap perubahan nilai *bit error rate*. Untuk pengujian sistem, akan dikirimkan 2160 bit kemudian dilihat nilai errornya. Pengujian sistem FHSS dengan *jamming* menggunakan parameter frekuensi pengujian berdasarkan standart IEEE 802.11 g, dalam pengujian dilakukan pada *channel* 5 dengan frekuensi pada *channel* 5 itu adalah 2.432 Ghz.

#### 4.2.1 Frekuensi Sinyal *Jamming* sama dengan Frekuensi Kerja pada Gangguan *Single Tone Jamming*

Pada pengujian sistem FHSS dengan gangguan *Single Tone Jamming* akan diuji dengan menggunakan frekuensi *jamming* sama dengan frekuensi kerja yaitu 2.432 GHz, serta dengan menggunakan nilai  $P_{tx}$  yang sama antara sistem FHSS dan 4-ary FSK. Hasil pengujian pada frekuensi *jamming* sama dengan frekuensi kerja dapat ditunjukkan pada Tabel 4.2 dan Gambar 4.2.

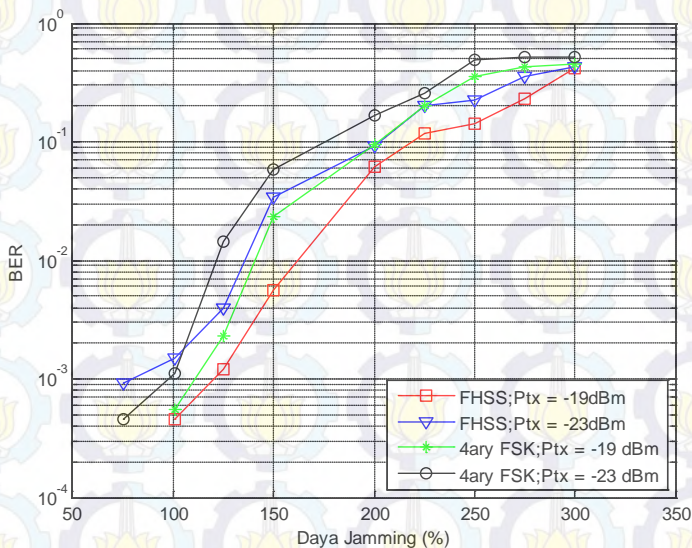
**Tabel 4.2** Nilai BER vs Daya *Single Tone Jamming* untuk Frekuensi *jamming* sama dengan Frekuensi kerja

Daya <i>Jamming</i> (%)	FHSS		4-ary FSK	
	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm
10	0	0	0	0
30	0	0	0	0
50	0	0	0	0
75	0	0.00092593	0	0.00046296
100	0.00046296	0.0015	0.000556	0.0011
125	0.0012	0.0039	0.0023	0.0144
150	0.0056	0.0339	0.0231	0.0586
200	0.0623	0.0916	0.0944	0.1689
225	0.1171	0.2034	0.1989	0.2578
250	0.1432	0.2218	0.3578	0.4856
275	0.2301	0.3569	0.4333	0.5178
300	0.4194	0.4235	0.4567	0.5222

Berdasarkan Tabel 4.2 dan Gambar 4.2 nilai BER yang dihasilkan saat menggunakan frekuensi *jamming* sama dengan frekuensi kerja mengalami jumlah *Bit Error Rate* yang besar. Dilihat berdasarkan daya pancar, bahwa disaat nilai daya pancar yang digunakan besar yaitu -19 dBm maka akan menghasilkan nilai BER yang kecil dan sebaliknya untuk nilai daya pancar yang kecil. Pada hasil pengujian frekuensi *jamming* sama dengan frekuensi kerja ini, nilai BER yang didapat akan



semakin besar jika nilai daya *jamming* yang digunakan semakin besar juga. Hasil pengujian dari frekuensi *jamming* sama dengan frekuensi kerja ini, sistem FHSS dan 4-ary FSK tidak terpengaruh oleh *jamming* saat daya sinyal *jamming* 75% sampai 10% untuk nilai  $P_{tx} = -19$  dBm dan untuk  $P_{tx} = -23$  dBm tidak terpengaruh oleh *jamming* saat daya sinyal *jamming* 50% sampai 10%. Berdasarkan hasil pengujian ini bahwa perbandingan hasil pengujian dari sistem FHSS lebih baik dari pada sistem 4-ary FSK.



**Gambar 4.2** Grafik BER vs Daya Sinyal *Jamming* pada Frekuensi kerja sama dengan Frekuensi *jamming* saat *Single Tone Jamming*

#### 4.2.2 Frekuensi Sinyal *Jamming* Tidak sama dengan Frekuensi Kerja pada pengujian *Single Tone Jamming*

Selain pengujian *single tone jamming* dengan menggunakan frekuensi kerja sama dengan frekuensi *jamming*, dilakukan juga pengukuran terhadap frekuensi *jamming* tidak sama dengan frekuensi kerja. Pada pengujian sistem FHSS dengan gangguan *Single Tone Jamming* saat menggunakan frekuensi *jamming* tidak sama dengan frekuensi kerja, frekuensi yang digunakan adalah 2.433 GHz. Nilai daya

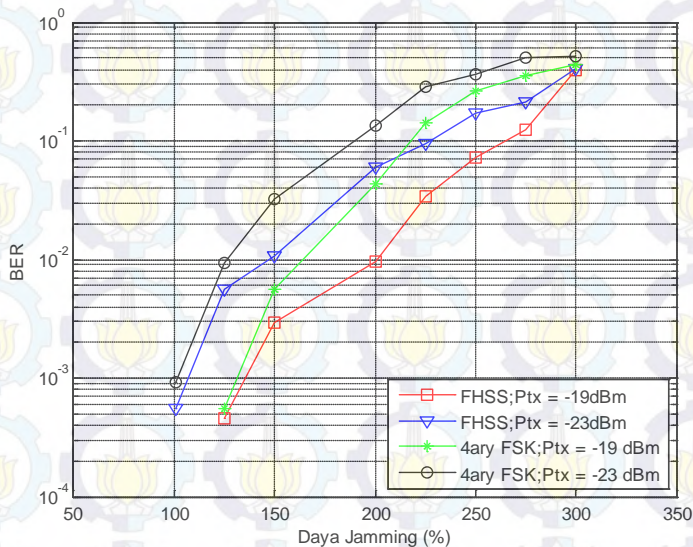
pancar yang digunakan sama dengan pengujian terhadap *Single Tone Jamming* menggunakan frekuensi *jamming* sama dengan frekuensi kerja yaitu  $P_{tx} = -19$  dBm dan  $P_{tx} = -23$  dBm, baik pada sistem FHSS maupun terhadap 4-ary FSK. Hasil pengujian dengan frekuensi *jamming* tidak sama dengan frekuensi kerja, dapat ditunjukkan pada Tabel 4.3 dan Gambar 4.3.

**Tabel 4.3** Nilai BER vs Daya *Single Tone Jamming* untuk Frekuensi *jamming* tidak sama dengan Frekuensi kerja

Daya <i>Jamming</i> (%)	FHSS		4-ary FSK	
	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm
10	0	0	0	0
30	0	0	0	0
50	0	0	0	0
75	0	0	0	0
100	0	0.000556	0	0.0004651
125	0.00046296	0.0056	0.00056	0.0093
150	0.0029	0.0106	0.0056	0.0323
200	0.0097	0.0606	0.0431	0.1356
225	0.0338	0.0957	0.1422	0.2821
250	0.0716	0.1729	0.2644	0.3611
275	0.1253	0.213	0.3567	0.4967
300	0.3927	0.4021	0.4367	0.5133

Hasil pengujian terhadap *Single Tone Jamming* menggunakan frekuensi *jamming* tidak sama dengan frekuensi kerja, dapat dilihat pada Tabel 4.3 dan Gambar 4.3. Pengujian ini, menghasilkan nilai BER yang lebih kecil dari pada nilai BER terhadap pengujian menggunakan frekuensi *jamming* sama dengan frekuensi kerja. Saat nilai daya *jamming* 300 % menghasilkan nilai BER untuk sistem FHSS pada  $P_{tx} = -19$  dBm sebesar 0.3927 dan saat  $P_{tx} = -23$  dBm menghasilkan nilai

BER sebesar 0.4021, sedangkan pada hasil pengujian frekuensi *jamming* sama dengan frekuensi kerja nilai BER untuk sistem FHSS saat  $P_{tx} = -19$  dBm sebesar 0.4194 dan  $P_{tx} = -23$  dBm sebesar 0.4235. Nilai BER untuk 4-ary FSK juga akan semakin kecil dibandingkan dengan pengujian terhadap frekuensi *jamming* sama dengan frekuensi kerja. Tetapi untuk pengaruh daya *jamming* tidak berbeda, yaitu semakin besar nilai daya *jamming* yang dilakukan maka akan semakin besar juga nilai BER yang dihasilkan. Pada hasil pengujian terhadap frekuensi *jamming* tidak sama dengan frekuensi kerja, saat nilai  $P_{tx} = -19$  dBm nilai BER yang dihasilkan tidak terpengaruh oleh *Single Tone Jamming* pada daya *jamming* 100% sampai 10% sedangkan untuk  $P_{tx} = -23$  dBm nilai yang dihasilkan tidak terpengaruh oleh gangguan *Single Tone Jamming* pada daya sinyal *jamming* 75% sampai dengan 10% .



**Gambar 4.3** Grafik BER vs Daya Sinyal *Jamming* pada Frekuensi kerja tidak sama dengan Frekuensi *jamming* saat *Single Tone Jamming*



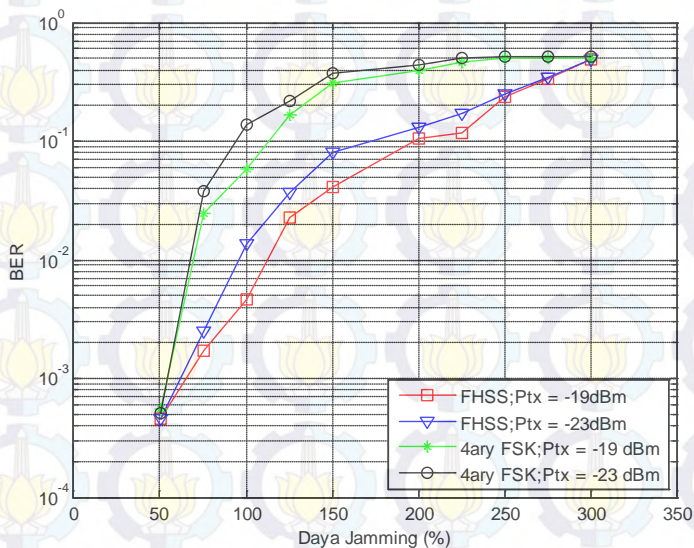
### 4.2.3 Frekuensi Sinyal *Jamming* sama dengan Frekuensi Kerja pada pengujian *Multi Tone Jamming*

Selain melakukan pengujian sistem FHSS dengan *Single Tone Jamming*, sistem juga diuji dengan *Multi Tone Jamming*. Parameter yang digunakan sama seperti melakukan pengujian dengan *Single Tone Jamming*, yaitu terhadap daya pancar, daya *jamming*, dan frekuensi *jamming*. Tetapi untuk *Multi Tone Jamming* menggunakan dua frekuensi *jamming*. Untuk pengaturan parameter-parameter yang lain sama seperti saat melakukan pengujian dengan *Single Tone Jamming*. Frekuensi *jamming* disesuaikan dengan salah satu dari frekuensi kerja, tetapi untuk pengujian kali ini frekuensi *jamming* satunya dibawah frekuensi kerja. Frekuensi *jamming* yang pertama menggunakan frekuensi pada 2.431 GHz dan frekuensi *jamming* kedua menggunakan fekuensi pada 2.432 GHz. Hasil pengujian ini dapat ditunjukkan pada Tabel 4.4 dan Gambar 4.4.

**Tabel 4.1** Nilai BER vs Daya *Multi Tone Jamming* untuk Frekuensi *jamming* sama dengan Frekuensi kerja

Daya <i>Jamming</i> (%)	FHSS		4-ary FSK	
	Ptx = -19 dBm	Ptx = -23 dBm	Ptx = -19 dBm	Ptx = -23 dBm
10	0	0	0	0
30	0	0	0	0
50	0.00046296	0.00046296	0.0005556	0.00051282
75	0.0017	0.0022	0.0244	0.0377
100	0.0047	0.0036	0.0588	0.1389
125	0.0226	0.0271	0.1689	0.2189
150	0.0411	0.0798	0.3111	0.3722
200	0.1056	0.1326	0.3912	0.4367
225	0.1162	0.1528	0.4644	0.4967
250	0.2359	0.2496	0.5014	0.5111
275	0.3382	0.3457	0.5067	0.5167
300	0.4892	0.4953	0.5078	0.52

Berdasarkan Tabel 4.4 dan Gambar 4.4 terdapat nilai BER yang lebih besar dibandingkan dengan hasil pengujian *Single Tone Jamming* saat menggunakan frekuensi *jamming* sama dengan frekuensi kerja, yaitu misalnya pada sistem FHSS saat  $P_{tx} = -19$  dBm dan daya *jamming* sebesar 300% menghasilkan nilai BER sebesar 0.4892 sedangkan untuk *Single Tone Jamming* menghasilkan nilai BER sebesar 0.4194. Hal ini karena *tone* yang digunakan sebanyak dua buah dan juga pengaruh frekuensi yang digunakan. Berdasarkan grafik pada Gambar 4.4 dapat disimpulkan bahwa semakin besar nilai daya *jamming* maka nilai bit yang salah akan semakin besar. Nilai kesalahan bit sistem FHSS lebih kecil dari sistem 4-ary FSK. Pengujian pada *Multi Tone Jamming* ini, sistem FHSS dan 4-ary FSK pada daya *jamming* sebesar 50% ke bawah tidak dipengaruhi oleh gangguan *jamming*.



**Gambar 4.1** Grafik BER vs Daya Sinyal *Jamming* pada Frekuensi kerja sama dengan Frekuensi *jamming* saat *Multi Tone Jamming*

#### 4.2.4 Frekuensi Sinyal *Jamming* Tidak sama dengan Frekuensi Kerja pada pengujian *Multi Tone Jamming*

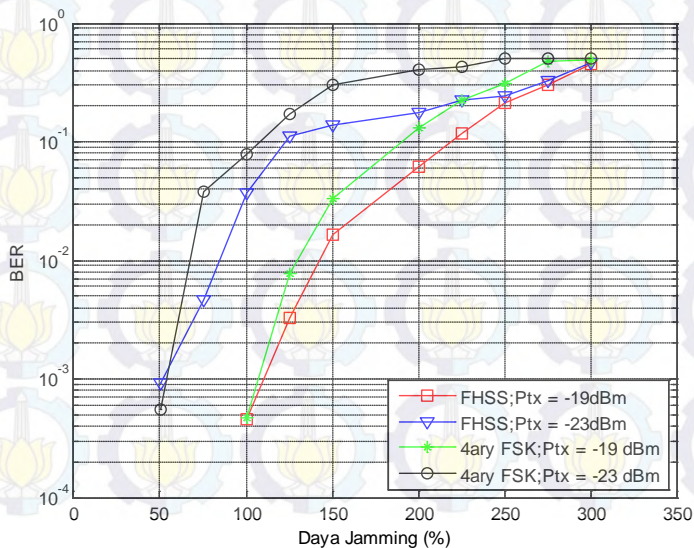
Pengujian ini dilakukan pada *Multi Tone Jamming* dengan frekuensi *jamming* tidak sama dengan frekuensi kerja. Pada pengujian ini, frekuensi *jamming* yang pertama menggunakan frekuensi 2.431 GHz dan frekuensi *jamming* kedua menggunakan frekuensi pada 2.433 GHz. Untuk parameter yang digunakan dalam pengujian ini sama dengan parameter pengujian pada *Single Tone Jamming* maupun pada pengujian *Multi Tone Jamming* dengan frekuensi *jamming* sama dengan frekuensi kerja, seperti parameter yang digunakan  $P_{tx} = -19$  dBm,  $P_{tx} = -23$  dBm dan nilai daya *jamming* yang digunakan terhadap pengukuran. Hasil pengujian ini dapat ditunjukkan pada Tabel 4.5 dan Gambar 4.5.

**Tabel 4.2** Nilai BER vs Daya *Multi Tone Jamming* untuk Frekuensi *jamming* tidak sama dengan Frekuensi kerja

Amplitudo (%)	FHSS		4-ary FSK	
	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm
10	0	0	0	0
30	0	0	0	0
50	0	0.00092593	0	0.0005556
75	0	0.0047	0	0.0377
100	0.00046296	0.0373	0.00046512	0.0778
125	0.0033	0.1123	0.0077	0.1733
150	0.0166	0.1371	0.0334	0.3033
200	0.0620	0.1764	0.1322	0.4044
225	0.1171	0.2218	0.2267	0.4300
250	0.2105	0.2407	0.3111	0.4989
275	0.3012	0.3269	0.4722	0.5044
300	0.4524	0.4594	0.4844	0.5067



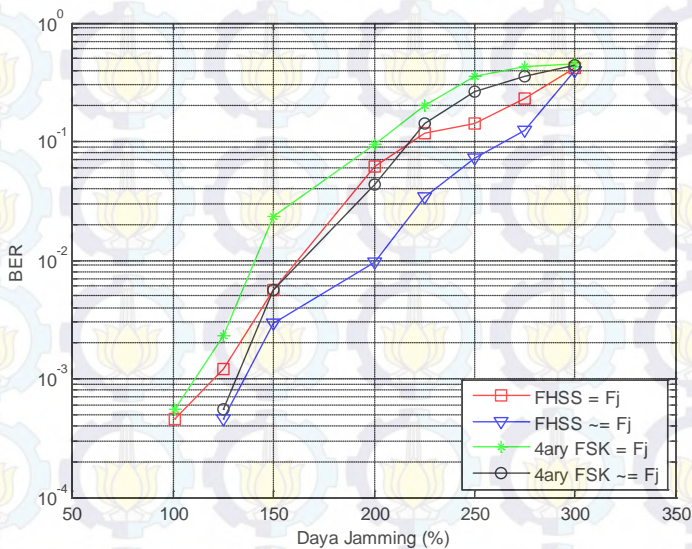
Pada hasil pengujian terhadap *Multi Tone Jamming* dengan frekuensi *jamming* tidak sama dengan frekuensi kerja hasil BER nya terlihat pada Tabel 4.5 dan grafiknya terlihat pada Gambar 4.5. Hasil pengujian yang dilakukan bahwa frekuensi *jamming* tidak sama dengan frekuensi kerja, sehingga menghasilkan nilai BER yang lebih kecil dari pada nilai BER *Multi Tone Jamming* dengan frekuensi *jamming* sama dengan frekuensi kerja. Nilai BER pada sistem FHSS saat daya *jamming* 300% dan  $P_{tx} = -19$  dBm untuk *Multi Tone Jamming* dengan frekuensi *jamming* tidak sama dengan frekuensi kerja menghasilkan BER sebesar 0.4524, sedangkan untuk *Multi Tone Jamming* pada frekuensi *jamming* sama dengan frekuensi kerja dengan parameter yang sama pada *Multi Tone Jamming* saat frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.4892, begitu juga untuk hasil yang saat nilai  $P_{tx} = -23$  dBm dengan daya *jamming* yang sama akan menghasilkan nilai BER yang lebih kecil juga. Hasil pengujian terhadap frekuensi *jamming* tidak sama dengan frekuensi kerja, pada sistem FHSS dan 4-ary FSK untuk nilai  $P_{tx} = -19$  dBm tidak dipengaruhi oleh gangguan *Multi Tone Jamming* saat daya *jamming* sebesar 75% sampai dengan 10%. Sedangkan untuk  $P_{tx} = -23$  dBm tidak dipengaruhi oleh gangguan *Multi Tone Jamming* pada 30% sampai 10%.



**Gambar 4.2** Grafik BER vs Daya Sinyal *Jamming* pada Frekuensi kerja tidak sama dengan Frekuensi *jamming* saat *Multi Tone Jamming*

### 4.3 Analisis Sistem

Hasil yang telah dilakukan dari berbagai macam model pengujian pada sistem FHSS dan 4-ary FSK sebagai perbandingan, maka dapat dianalisa kinerja dari sistem FHSS untuk melihat ketahanan terhadap *jamming* dengan membandingkan nilai BER terhadap daya *jamming*. Analisa kinerja sistem FHSS dengan gangguan *Single Tone Jamming* dapat ditunjukkan pada Gambar 4.6 dan Gambar 4.7.



**Gambar 4.3** Grafik BER vs Daya Sinyal *Jamming* pada Gangguan *Single Tone Jamming* saat  $P_{tx} = -19$  dBm

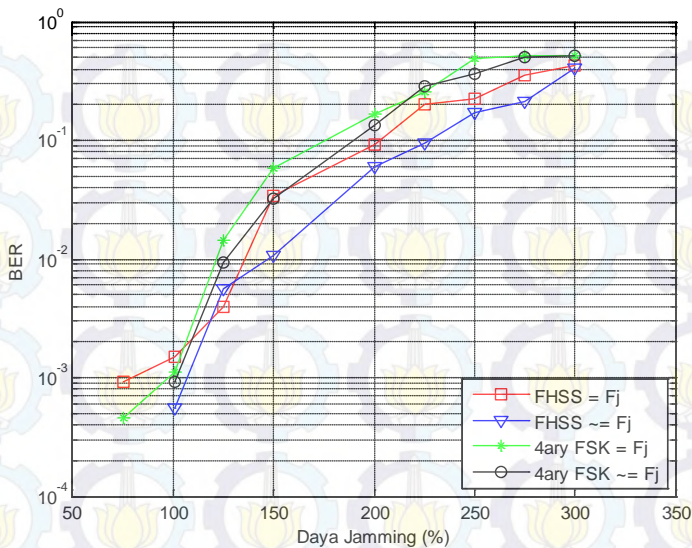
Berdasarkan dari grafik Gambar 4.6 merupakan perbandingan nilai BER sistem FHSS dengan 4-ary FSK pada frekuensi *jamming* sama dengan frekuensi kerja dan frekuensi *jamming* tidak sama dengan frekuensi kerja pada *Single Tone Jamming* saat  $P_{tx} = -19$  dBm. Setiap kenaikan nilai daya *jamming* maka nilai BER yang dihasilkan akan semakin besar juga, dari grafik terlihat bahwa nilai BER saat frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER lebih

besar dari pada frekuensi *jamming* tidak sama dengan frekuensi kerja. Untuk frekuensi *jamming* sama dengan frekuensi kerja dengan daya *jamming* sebesar 300% menghasilkan nilai BER sebesar 0.4194, sedangkan untuk frekuensi *jamming* tidak sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.3927. Jadi dari penjelasan tersebut bahwa frekuensi *jamming* tidak sama dengan frekuensi kerja lebih baik dibandingkan dengan frekuensi *jamming* sama dengan frekuensi kerja. Pada sistem 4-ary FSK menggunakan daya *jamming* saat 300% juga menghasilkan nilai BER sebesar 0.4567 untuk frekuensi *jamming* sama dengan frekuensi kerja, sedangkan pada frekuensi *jamming* tidak sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.4367. Semakin besar nilai daya *jamming* yang digunakan maka akan semakin besar juga nilai BER yang dihasilkan. Berdasarkan Gambar 4.6 merupakan hasil pengujian untuk gangguan *Single Tone Jamming* saat  $P_{tx} = -19$  dBm pada sistem FHSS dan 4-ary FSK, saat frekuensi *Jamming* sama dengan frekuensi kerja tidak mengalami gangguan *Single Tone Jamming* pada daya sinyal *jamming* 75% ke bawah. Sedangkan untuk  $P_{tx} = -19$  dBm saat frekuensi *jamming* tidak sama dengan frekuensi kerja tidak mengalami gangguan *Single Tone Jamming* pada nilai daya *jamming* dari 100% sampai 10%, baik pada sistem FHSS maupun pada 4-ary FSK.

Pada Gambar 4.7 merupakan hasil pengujian dari sistem FHSS dan 4-ary FSK dengan *Single Tone Jamming* saat  $P_{tx} = -23$  dBm. Perbandingan dari Gambar 4.7 dengan Gambar 4.6, nilai BER untuk Gambar 4.6 lebih baik, karena semakin besar nilai  $P_{tx}$  yang digunakan akan menghasilkan nilai BER yang semakin kecil. Pada Gambar 4.7 juga menggambarkan bahwa nilai BER saat diganggu frekuensi kerja oleh frekuensi *jamming* menghasilkan nilai BER yang semakin besar dari pada frekuensi kerja yang tidak diganggu oleh frekuensi *jamming*. Hasil nilai BER untuk sistem FHSS dan 4-ary FSK pada  $P_{tx} = -23$  dBm saat frekuensi *jamming* sama dengan frekuensi kerja tidak terpengaruh oleh gangguan *Single Tone Jamming* pada daya *jamming* 50% ke bawah, sedangkan saat frekuensi *jamming* tidak sama dengan frekuensi *jamming* nilai BER tidak berpengaruh oleh gangguan *Single Tone Jamming* pada daya *jamming* 75% sampai 10% untuk sistem FHSS dan 4-ary FSK.

Berdasarkan Gambar 4.8 dan Gambar 4.9 merupakan hasil pengujian dari *Multi Tone Jamming* baik dengan frekuensi *jamming* sama dengan frekuensi kerja maupun frekuensi *jamming* tidak sama dengan frekuensi kerja.

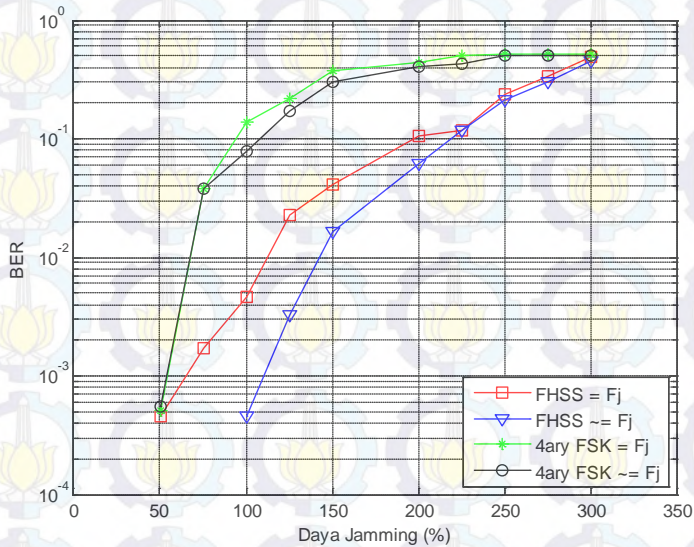




**Gambar 4.4** Grafik BER vs Daya Sinyal *Jamming* pada Gangguan *Single Tone Jamming* saat  $P_{tx} = -23$  dBm

Pada Gambar 4.8 menunjukkan hasil dari *Multi Tone Jamming* pada saat  $P_{tx}$  bernilai -19 dBm. Nilai BER yang didapat pada  $P_{tx}$  bernilai -19 dBm ini, menggunakan parameter yang tidak berbeda dengan parameter pengujian pada *Single Tone Jamming*, tetapi pada *Multi Tone Jamming* menggunakan dua frekuensi *jamming*. Hasil dari *Multi Tone Jamming* saat  $P_{tx} = -19$  dBm pada frekuensi *jamming* sama dengan frekuensi kerja tidak terpengaruh oleh gangguan *jamming* pada daya *jamming* 30% ke bawah, sedangkan untuk frekuensi *jamming* tidak sama dengan frekuensi kerja tidak terpengaruh oleh gangguan *jamming* pada daya *jamming* 75% ke bawah. Hasil untuk 4-ary FSK tidak terpengaruh oleh gangguan *jamming* pada daya *jamming* 30% ke bawah, baik untuk kerja frekuensi *jamming* sama dan tidak sama dengan frekuensi kerja. Terlihat dari grafik, pada gangguan *Multi Tone Jamming* berkerja pada salah satu dari frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER yang lebih besar. Perbandingan nilai dari sistem FHSS lebih baik dari pada sistem 4-ary FSK, baik yang secara

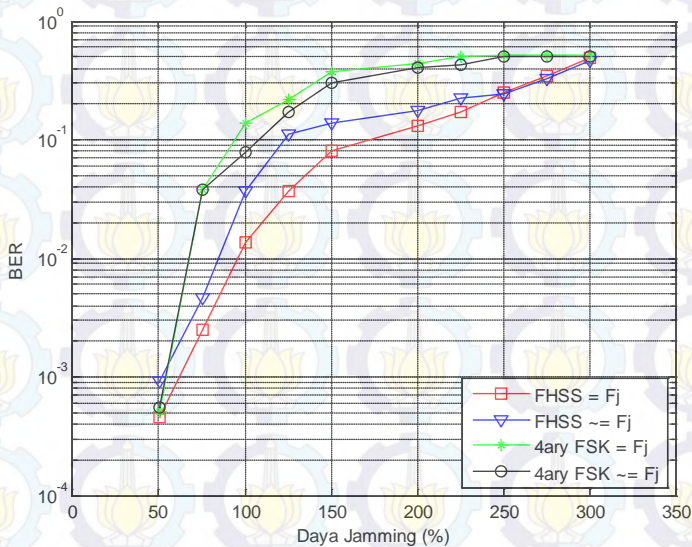
frekuensi *jamming* sama dengan frekuensi kerja maupun saat frekuensi *jamming* tidak sama dengan frekuensi kerja.



**Gambar 4.5** Grafik BER vs Daya Sinyal *Jamming* pada *Multi Tone Jamming* saat  $P_{tx} = -19$  dBm

Pada Gambar 4.9 merupakan grafik BER saat *Multi Tone Jamming* menggunakan nilai  $P_{tx}$  sebesar -23 dBm. Berdasarkan perbandingan nilai BER saat menggunakan nilai  $P_{tx}$  -23 dBm lebih besar dari nilai  $P_{tx}$  bernilai -19 dBm, hal ini karena nilai daya pancar yang kecil akan menghasilkan nilai BER yang besar dan sebaliknya untuk nilai daya pancar yang besar menghasilkan nilai BER yang kecil. Nilai BER saat daya *jamming* 300% untuk  $P_{tx} = -23$  dBm misalnya untuk frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.4953 sedangkan untuk  $P_{tx} = -19$  dBm sebesar 0.4892. Hasil pengujian menunjukkan bahwa saat frekuensi *jamming* berkerja pada frekuensi kerja akan berpengaruh besar terhadap nilai BER yang dihasilkan baik pada *Multi Tone Jamming* maupun pada *Single Tone Jamming*. Saat frekuensi kerjanya di pengaruhi oleh frekuensi *jamming* akan menghasilkan nilai BER akan semakin besar. Begitu juga terhadap nilai daya *jamming* yang digunakan, akan menghasilkan nilai BER yang

semakin besar jika daya *jamming* yang digunakan semakin besar juga. Hasil  $P_{tx} = -23$  dBm untuk hasil sistem FHSS dan 4-ary FSK saat frekuensi *jamming* sama dan tidak sama dengan frekuensi kerja tidak dipengaruhi oleh gangguan *Multi Tone Jamming* pada daya *jamming* sebesar 30% sampai dengan 10%.



**Gambar 4. 6** Grafik BER vs Daya Sinyal *Jamming* pada *Multi Tone Jamming* saat  $P_{tx} = -23$  dBm

Berdasarkan hasil pengujian yang dilakukan pada sistem FHSS ini lebih baik dari pada sistem 4-ary FSK, untuk nilai sistem FHSS yang menggunakan gangguan *Single Tone Jamming* menghasilkan nilai BER yang lebih baik dari pada menggunakan gangguan *Multi Tone Jamming*. Hal ini karena, pada *Multi Tone Jamming* menggunakan dua frekuensi *jamming*. Dalam hasil yang didapat, nilai daya *jamming* juga berpengaruh terhadap nilai BER yang dihasilkan, apabila semakin besar nilai daya *jamming* maka akan menghasilkan nilai BER yang semakin besar juga dan begitu sebaliknya untuk nilai daya *jamming* yang digunakan kecil. Dalam pengujian ini, sistem FHSS tidak mengalami gangguan *Single Tone Jamming* saat frekuensi *jamming* sama dengan frekuensi pada daya *jamming* 75% ke bawah untuk nilai  $P_{tx} = -19$  dBm



dan saat  $P_{tx} = -23$  dBm tidak terpengaruh pada daya *jamming* 50% ke bawah. Sedangkan untuk *Single Tone Jamming* pada frekuensi *jamming* tidak sama dengan frekuensi kerja tidak mengalami gangguan saat daya *jamming* 100% ke bawah untuk nilai  $P_{tx} = -19$  dBm dan  $P_{tx} = -23$  dBm pada daya *jamming* 75% ke bawah. Hasil pengujian dari gangguan *Multi Tone Jamming* pada frekuensi *jamming* sama dengan frekuensi kerja tidak berpengaruh terhadap gangguan jamming untuk  $P_{tx} = -19$  dBm dan  $P_{tx} = -23$  dBm pada daya *jamming* 30% ke bawah. *Multi Tone Jamming* untuk frekuensi *jamming* tidak sama dengan frekuensi kerja pada  $P_{tx} = -19$  dBm tidak terpengaruh gangguan *jamming* saat daya *jamming* 75% ke bawah, sedangkan  $P_{tx} = -23$  dBm pada daya *jamming* 30% ke bawah. Untuk nilai  $P_{tx}$  yang digunakan semakin besar maka menghasilkan nilai BER yang semakin kecil. Jadi dari hasil pengujian ini yang dapat mempengaruhi nilai BER pada sistem FHSS ini adalah daya *jamming* dan nilai daya pancar yang digunakan.



## BAB V PENUTUP

### 5.1 Kesimpulan

Berdasarkan dari hasil proses implementasi, pengujian, dan analisis dapat ditarik kesimpulan sebagai berikut :

1. Hasil implementasi untuk sistem FHSS dan 4-ary FSK tanpa *jamming*, dipengaruhi oleh nilai Tx Rf jika semakin besar Tx Rf yang dipancarkan maka akan menghasilkan nilai BER yang semakin kecil.
2. Hasil implementasi sistem FHSS dan 4-ary FSK dengan menggunakan *jamming* berpengaruh terhadap nilai daya *jamming* yang digunakan, semakin besar daya *jamming* maka akan semakin besar juga nilai BER yang dihasilkan. Misalnya saat frekuensi *jamming* sama dengan frekuensi kerja pada *Single Tone Jamming* dengan daya *jamming* 300% menghasilkan nilai BER sebesar 0.4194 dan saat daya *jamming* 100% menghasilkan nilai BER sebesar 0.00046296.
3. Hasil pengujian terhadap *Single Tone Jamming* lebih baik dari pada *Multi Tone Jamming*, pada *Single Tone Jamming* menghasilkan nilai BER saat daya *jamming* 300% sebesar 0.4194 sedangkan pada *Multi Tone Jamming* saat daya *jamming* 300% juga menghasilkan nilai BER sebesar 0.4892.
4. Jika melakukan *jamming* pada frekuensi kerja akan mengakibatkan nilai BER yang dihasilkan akan semakin besar, karena dapat mengganggu sistem kerja pada frekuensi kerja.
5. Hasil implementasi pada gangguan *Single Tone Jamming* untuk frekuensi *jamming* sama dengan frekuensi kerja tidak terpengaruh oleh *jamming* saat kondisi  $P_{tx} = -19$  dBm dengan daya *jamming* 75% dan  $P_{tx} = -23$  dBm pada daya *jamming* 50%, sedangkan untuk frekuensi *jamming* tidak sama dengan frekuensi kerja pada  $P_{tx} = -19$  dBm saat daya *jamming* 100% dan  $P_{tx} = -23$  dBm dengan daya *jamming* 75%.
6. Hasil dengan *Multi Tone Jamming* untuk frekuensi *jamming* sama dengan frekuensi kerja tidak di pengaruhi oleh *jamming* pada nilai daya *jamming* 30% untuk  $P_{tx} = -19$  dBm dan  $P_{tx} = -23$  dBm, sedangkan pada frekuensi *jamming* tidak sama dengan

frekuensi kerja pada  $P_{tx} = -19$  dBm dengan daya *jamming* 75% dan  $P_{tx} = -23$  dBm dengan daya *jamming* 30%.

## 5.2 Saran

Saran yang dapat diajukan untuk penelitian lebih lanjut mengenai topik ini adalah :

1. Pengujian sistem FHSS ini dapat dilakukan terhadap berbagai teknik *jamming* yang lain, misalnya *Partial Band Noise Jamming* (PBNJ), *Broadband Noise Jamming* (BBNJ) atau *Pulse Jamming*.
2. Jika implementasi selanjutnya dapat dilakukan menggunakan frekuensi tinggi agar mendapatkan hasil yang maksimal.



## DAFTAR PUSTAKA

- [1] Goldsmith, Andrea. *"Wireless Communications"*. Cambridge University Press. 2005.
- [2] N, Kafadarova., Mileva N., dan Stoyonova S., *"Remote Wireless Communications Lab in Real Time"*. IEEE Global Engineering Education Conference (EDUCUN). 2013.
- [3] Hodgdon, Charles, *"Adaptive Frequency Hopping for Reduced Interference between Bluetooth® and Wireless LAN"*, Ericsson Technology Licensing, 2003.
- [4] Torrieri, Don, *"Principles of Spread-Spectrum Communication Systems"*, Springer, 2005.
- [5] Olšovský, Peter, *"Design And Simulation Of Frequency Hopping Technique In Matlab"*, Slovak University of Technology.
- [6] Poisel, Richard, *"Modern Communications Jamming Principles and Techniques"*, Second Edition, Artech House, 2011.
- [7] Michael, Knox dan Elza Erkip, *"Implementation of Cooperative Communication using Software Defined Radios"*, Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on.
- [8] Murphy, Patrick. Ashu Sabharwal, dan Behnaam Aazhang, *"Design Of WARP: A Wireless Open-Access Research Platform"*, 14<sup>th</sup> European Signal Processing Conference (EUSIPCO 2006), Florence, Italy.
- [9] Physical Layer Prototyping using WARPLab. [Online] November 30, 2014. [Disunting: Desember 01, 2014.] [http://warpproject.org/trac/wiki/Workshops/Rice\\_2008](http://warpproject.org/trac/wiki/Workshops/Rice_2008)
- [10] WARPLab 6 Framework Overview. *WARP Project - Wireless Open Access Research Platform*. [Online] September 2014. [Disunting: Desember 04, 2014.] <http://warpproject.org/trac/wiki/WARPLab6>
- [11] Maxim Integrated Product Inc. MAX2828/MAX2829 Single-Dual-Band 802.11a/b/g World-band Transceiver ICs. [Online] Oktober 16, 2014. [Disunting: Desember 11, 2014.] <http://datasheets.maximintegrated.com/en/ds/MAX2828-MAX2829.pdf>
- [12] A, Madhu, *"DSP Implementation of OFDM Acoustic Modem"*, Master of Engineering Thesis : Department of Electronics &

Communication Engineering ; National Institute of Technology,  
Rourkela, 2008.

- [13] Yerl, Seung Lee, “*An IEEE 802.11g WLAN Digital Baseband Processor Using Hybrid Channel Estimation for Wireless Home A/V Receivers*”, IEEE Transactions on Consumer Electronics, Vol. 55, No. 3, AUGUST 2009.
- [14] S, Ethan Hennessey, “*Opportunistic Access In Frequency Hopping Cognitive Radio Networks*”, Master of Engineering Thesis : Department of The Air Force Air University ; Air Force Institute of Technology, Maret, 2014.

### 4.2.3 Frekuensi Sinyal *Jamming* sama dengan Frekuensi Kerja pada pengujian *Multi Tone Jamming*

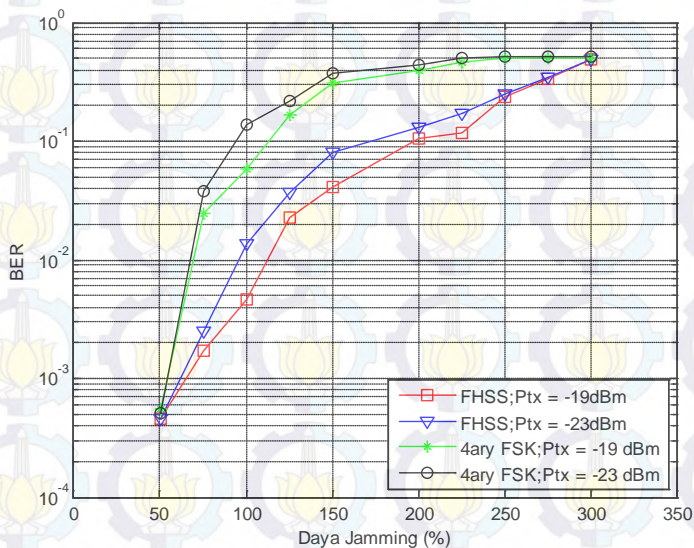
Selain melakukan pengujian sistem FHSS dengan *Single Tone Jamming*, sistem juga diuji dengan *Multi Tone Jamming*. Parameter yang digunakan sama seperti melakukan pengujian dengan *Single Tone Jamming*, yaitu terhadap daya pancar, daya *jamming*, dan frekuensi *jamming*. Tetapi untuk *Multi Tone Jamming* menggunakan dua frekuensi *jamming*. Untuk pengaturan parameter-parameter yang lain sama seperti saat melakukan pengujian dengan *Single Tone Jamming*. Frekuensi *jamming* disesuaikan dengan salah satu dari frekuensi kerja, tetapi untuk pengujian kali ini frekuensi *jamming* satunya dibawah frekuensi kerja. Frekuensi *jamming* yang pertama menggunakan frekuensi pada 2.431 GHz dan frekuensi *jamming* kedua menggunakan fekuensi pada 2.432 GHz. Hasil pengujian ini dapat ditunjukkan pada Tabel 4.4 dan Gambar 4.4.

**Tabel 4.1** Nilai BER vs Daya *Multi Tone Jamming* untuk Frekuensi *jamming* sama dengan Frekuensi kerja

Daya <i>Jamming</i> (%)	FHSS		4-ary FSK	
	Ptx = -19 dBm	Ptx = -23 dBm	Ptx = -19 dBm	Ptx = -23 dBm
10	0	0	0	0
30	0	0	0	0
50	0.00046296	0.00046296	0.0005556	0.00051282
75	0.0017	0.0022	0.0244	0.0377
100	0.0047	0.0036	0.0588	0.1389
125	0.0226	0.0271	0.1689	0.2189
150	0.0411	0.0798	0.3111	0.3722
200	0.1056	0.1326	0.3912	0.4367
225	0.1162	0.1528	0.4644	0.4967
250	0.2359	0.2496	0.5014	0.5111
275	0.3382	0.3457	0.5067	0.5167
300	0.4892	0.4953	0.5078	0.52



Berdasarkan Tabel 4.4 dan Gambar 4.4 terdapat nilai BER yang lebih besar dibandingkan dengan hasil pengujian *Single Tone Jamming* saat menggunakan frekuensi *jamming* sama dengan frekuensi kerja, yaitu misalnya pada sistem FHSS saat  $P_{tx} = -19$  dBm dan daya *jamming* sebesar 300% menghasilkan nilai BER sebesar 0.4892 sedangkan untuk *Single Tone Jamming* menghasilkan nilai BER sebesar 0.4194. Hal ini karena *tone* yang digunakan sebanyak dua buah dan juga pengaruh frekuensi yang digunakan. Berdasarkan grafik pada Gambar 4.4 dapat disimpulkan bahwa semakin besar nilai daya *jamming* maka nilai bit yang salah akan semakin besar. Nilai kesalahan bit sistem FHSS lebih kecil dari sistem 4-ary FSK. Pengujian pada *Multi Tone Jamming* ini, sistem FHSS dan 4-ary FSK pada daya *jamming* sebesar 50% ke bawah tidak dipengaruhi oleh gangguan *jamming*.



**Gambar 4.1** Grafik BER vs Daya Sinyal *Jamming* pada Frekuensi kerja sama dengan Frekuensi *jamming* saat *Multi Tone Jamming*

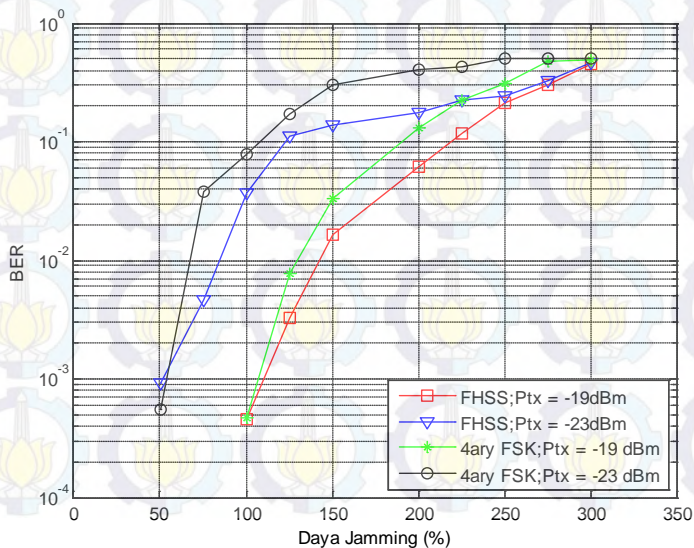
#### 4.2.4 Frekuensi Sinyal *Jamming* Tidak sama dengan Frekuensi Kerja pada pengujian *Multi Tone Jamming*

Pengujian ini dilakukan pada *Multi Tone Jamming* dengan frekuensi *jamming* tidak sama dengan frekuensi kerja. Pada pengujian ini, frekuensi *jamming* yang pertama menggunakan frekuensi 2.431 GHz dan frekuensi *jamming* kedua menggunakan frekuensi pada 2.433 GHz. Untuk parameter yang digunakan dalam pengujian ini sama dengan parameter pengujian pada *Single Tone Jamming* maupun pada pengujian *Multi Tone Jamming* dengan frekuensi *jamming* sama dengan frekuensi kerja, seperti parameter yang digunakan  $P_{tx} = -19$  dBm,  $P_{tx} = -23$  dBm dan nilai daya *jamming* yang digunakan terhadap pengukuran. Hasil pengujian ini dapat ditunjukkan pada Tabel 4.5 dan Gambar 4.5.

**Tabel 4.2** Nilai BER vs Daya *Multi Tone Jamming* untuk Frekuensi *jamming* tidak sama dengan Frekuensi kerja

Amplitudo (%)	FHSS		4-ary FSK	
	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm	$P_{tx} = -19$ dBm	$P_{tx} = -23$ dBm
10	0	0	0	0
30	0	0	0	0
50	0	0.00092593	0	0.0005556
75	0	0.0047	0	0.0377
100	0.00046296	0.0373	0.00046512	0.0778
125	0.0033	0.1123	0.0077	0.1733
150	0.0166	0.1371	0.0334	0.3033
200	0.0620	0.1764	0.1322	0.4044
225	0.1171	0.2218	0.2267	0.4300
250	0.2105	0.2407	0.3111	0.4989
275	0.3012	0.3269	0.4722	0.5044
300	0.4524	0.4594	0.4844	0.5067

Pada hasil pengujian terhadap *Multi Tone Jamming* dengan frekuensi *jamming* tidak sama dengan frekuensi kerja hasil BER nya terlihat pada Tabel 4.5 dan grafiknya terlihat pada Gambar 4.5. Hasil pengujian yang dilakukan bahwa frekuensi *jamming* tidak sama dengan frekuensi kerja, sehingga menghasilkan nilai BER yang lebih kecil dari pada nilai BER *Multi Tone Jamming* dengan frekuensi *jamming* sama dengan frekuensi kerja. Nilai BER pada sistem FHSS saat daya *jamming* 300% dan  $P_{tx} = -19$  dBm untuk *Multi Tone Jamming* dengan frekuensi *jamming* tidak sama dengan frekuensi kerja menghasilkan BER sebesar 0.4524, sedangkan untuk *Multi Tone Jamming* pada frekuensi *jamming* sama dengan frekuensi kerja dengan parameter yang sama pada *Multi Tone Jamming* saat frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.4892, begitu juga untuk hasil yang saat nilai  $P_{tx} = -23$  dBm dengan daya *jamming* yang sama akan menghasilkan nilai BER yang lebih kecil juga. Hasil pengujian terhadap frekuensi *jamming* tidak sama dengan frekuensi kerja, pada sistem FHSS dan 4-ary FSK untuk nilai  $P_{tx} = -19$  dBm tidak dipengaruhi oleh gangguan *Multi Tone Jamming* saat daya *jamming* sebesar 75% sampai dengan 10%. Sedangkan untuk  $P_{tx} = -23$  dBm tidak dipengaruhi oleh gangguan *Multi Tone Jamming* pada 30% sampai 10%.

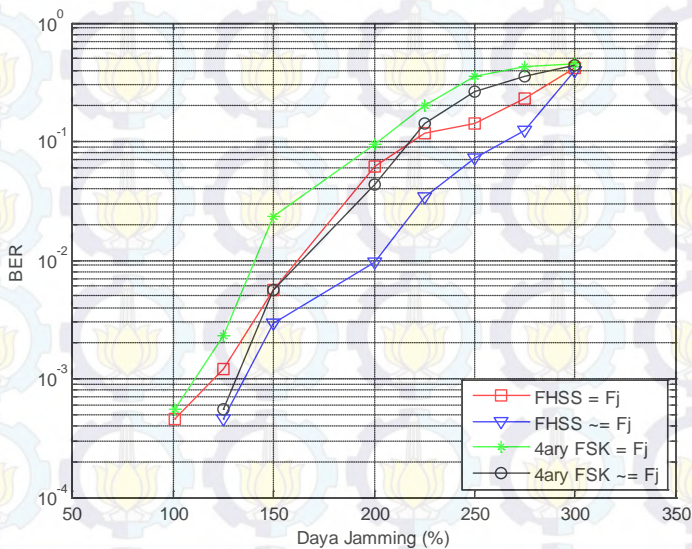




**Gambar 4.2** Grafik BER vs Daya Sinyal *Jamming* pada Frekuensi kerja tidak sama dengan Frekuensi *jamming* saat *Multi Tone Jamming*

### 4.3 Analisis Sistem

Hasil yang telah dilakukan dari berbagai macam model pengujian pada sistem FHSS dan 4-ary FSK sebagai perbandingan, maka dapat dianalisa kinerja dari sistem FHSS untuk melihat ketahanan terhadap *jamming* dengan membandingkan nilai BER terhadap daya *jamming*. Analisa kinerja sistem FHSS dengan gangguan *Single Tone Jamming* dapat ditunjukkan pada Gambar 4.6 dan Gambar 4.7.



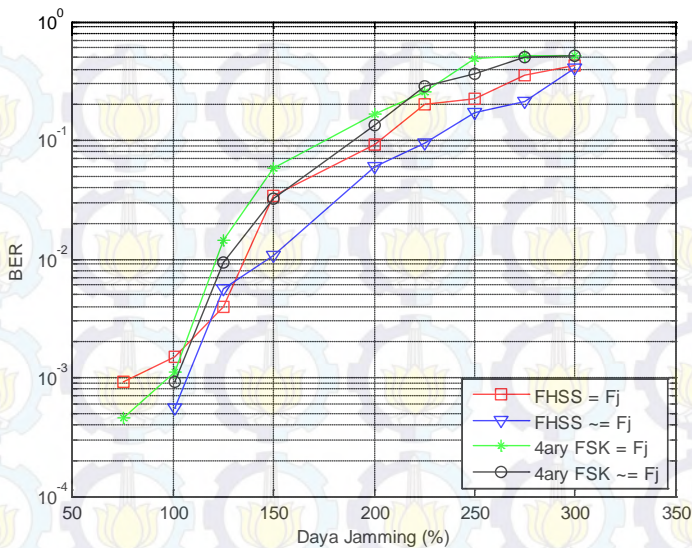
**Gambar 4.3** Grafik BER vs Daya Sinyal *Jamming* pada Gangguan *Single Tone Jamming* saat  $P_{tx} = -19$  dBm

Berdasarkan dari grafik Gambar 4.6 merupakan perbandingan nilai BER sistem FHSS dengan 4-ary FSK pada frekuensi *jamming* sama dengan frekuensi kerja dan frekuensi *jamming* tidak sama dengan frekuensi kerja pada *Single Tone Jamming* saat  $P_{tx} = -19$  dBm. Setiap kenaikan nilai daya *jamming* maka nilai BER yang dihasilkan akan semakin besar juga, dari grafik terlihat bahwa nilai BER saat frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER lebih

besar dari pada frekuensi *jamming* tidak sama dengan frekuensi kerja. Untuk frekuensi *jamming* sama dengan frekuensi kerja dengan daya *jamming* sebesar 300% menghasilkan nilai BER sebesar 0.4194, sedangkan untuk frekuensi *jamming* tidak sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.3927. Jadi dari penjelasan tersebut bahwa frekuensi *jamming* tidak sama dengan frekuensi kerja lebih baik dibandingkan dengan frekuensi *jamming* sama dengan frekuensi kerja. Pada sistem 4-ary FSK menggunakan daya *jamming* saat 300% juga menghasilkan nilai BER sebesar 0.4567 untuk frekuensi *jamming* sama dengan frekuensi kerja, sedangkan pada frekuensi *jamming* tidak sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.4367. Semakin besar nilai daya *jamming* yang digunakan maka akan semakin besar juga nilai BER yang dihasilkan. Berdasarkan Gambar 4.6 merupakan hasil pengujian untuk gangguan *Single Tone Jamming* saat  $P_{tx} = -19$  dBm pada sistem FHSS dan 4-ary FSK, saat frekuensi *Jamming* sama dengan frekuensi kerja tidak mengalami gangguan *Single Tone Jamming* pada daya sinyal *jamming* 75% ke bawah. Sedangkan untuk  $P_{tx} = -19$  dBm saat frekuensi *jamming* tidak sama dengan frekuensi kerja tidak mengalami gangguan *Single Tone Jamming* pada nilai daya *jamming* dari 100% sampai 10%, baik pada sistem FHSS maupun pada 4-ary FSK.

Pada Gambar 4.7 merupakan hasil pengujian dari sistem FHSS dan 4-ary FSK dengan *Single Tone Jamming* saat  $P_{tx} = -23$  dBm. Perbandingan dari Gambar 4.7 dengan Gambar 4.6, nilai BER untuk Gambar 4.6 lebih baik, karena semakin besar nilai  $P_{tx}$  yang digunakan akan menghasilkan nilai BER yang semakin kecil. Pada Gambar 4.7 juga menggambarkan bahwa nilai BER saat diganggu frekuensi kerja oleh frekuensi *jamming* menghasilkan nilai BER yang semakin besar dari pada frekuensi kerja yang tidak diganggu oleh frekuensi *jamming*. Hasil nilai BER untuk sistem FHSS dan 4-ary FSK pada  $P_{tx} = -23$  dBm saat frekuensi *jamming* sama dengan frekuensi kerja tidak terpengaruh oleh gangguan *Single Tone Jamming* pada daya *jamming* 50% ke bawah, sedangkan saat frekuensi *jamming* tidak sama dengan frekuensi *jamming* nilai BER tidak berpengaruh oleh gangguan *Single Tone Jamming* pada daya *jamming* 75% sampai 10% untuk sistem FHSS dan 4-ary FSK.

Berdasarkan Gambar 4.8 dan Gambar 4.9 merupakan hasil pengujian dari *Multi Tone Jamming* baik dengan frekuensi *jamming* sama dengan frekuensi kerja maupun frekuensi *jamming* tidak sama dengan frekuensi kerja.

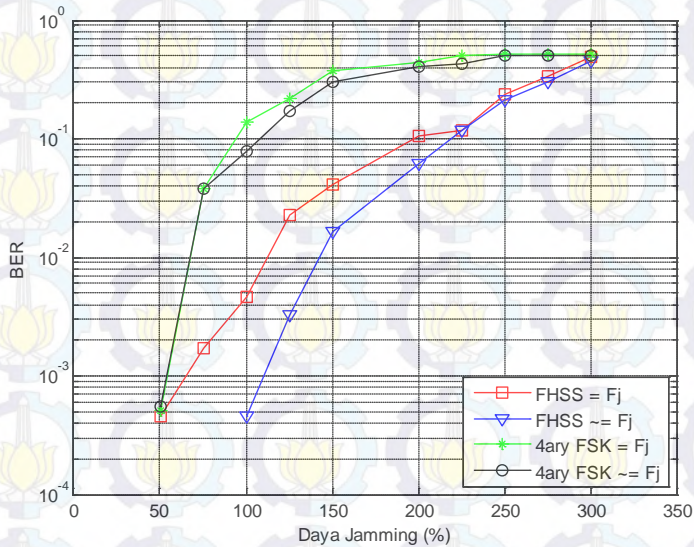


**Gambar 4.4** Grafik BER vs Daya Sinyal *Jamming* pada Gangguan *Single Tone Jamming* saat  $P_{tx} = -23$  dBm

Pada Gambar 4.8 menunjukan hasil dari *Multi Tone Jamming* pada saat  $P_{tx}$  bernilai -19 dBm. Nilai BER yang didapat pada  $P_{tx}$  bernilai -19 dBm ini, menggunakan parameter yang tidak berbeda dengan parameter pengujian pada *Single Tone Jamming*, tetapi pada *Multi Tone Jamming* menggunakan dua frekuensi *jamming*. Hasil dari *Multi Tone Jamming* saat  $P_{tx} = -19$  dBm pada frekuensi *jamming* sama dengan frekuensi kerja tidak terpengaruh oleh gangguan *jamming* pada daya *jamming* 30% ke bawah, sedangkan untuk frekuensi *jamming* tidak sama dengan frekuensi kerja tidak terpengaruh oleh gangguan *jamming* pada daya *jamming* 75% ke bawah. Hasil untuk 4-ary FSK tidak terpengaruh oleh gangguan *jamming* pada daya *jamming* 30% ke bawah, baik untuk kerja frekuensi *jamming* sama dan tidak sama dengan frekuensi kerja. Terlihat dari grafik, pada gangguan *Multi Tone Jamming* berkerja pada salah satu dari frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER yang lebih besar. Perbandingan nilai dari sistem FHSS lebih baik dari pada sistem 4-ary FSK, baik yang secara



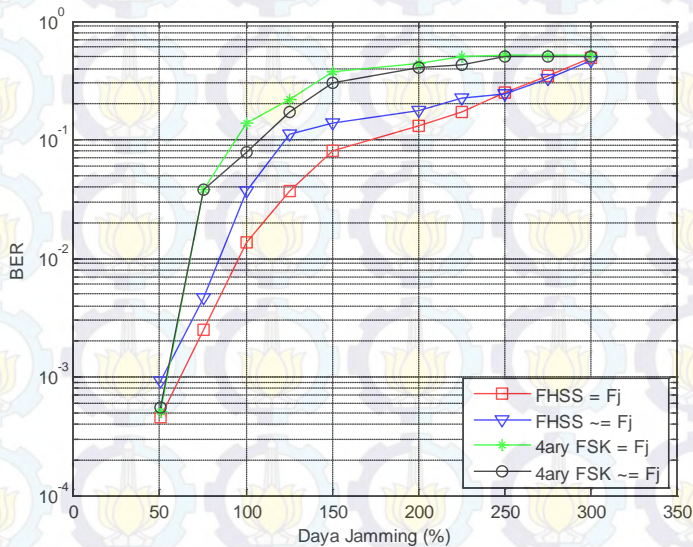
frekuensi *jamming* sama dengan frekuensi kerja maupun saat frekuensi *jamming* tidak sama dengan frekuensi kerja.



**Gambar 4.5** Grafik BER vs Daya Sinyal *Jamming* pada *Multi Tone Jamming* saat  $P_{tx} = -19$  dBm

Pada Gambar 4.9 merupakan grafik BER saat *Multi Tone Jamming* menggunakan nilai  $P_{tx}$  sebesar -23 dBm. Berdasarkan perbandingan nilai BER saat menggunakan nilai  $P_{tx}$  -23 dBm lebih besar dari nilai  $P_{tx}$  bernilai -19 dBm, hal ini karena nilai daya pancar yang kecil akan menghasilkan nilai BER yang besar dan sebaliknya untuk nilai daya pancar yang besar menghasilkan nilai BER yang kecil. Nilai BER saat daya *jamming* 300% untuk  $P_{tx} = -23$  dBm misalnya untuk frekuensi *jamming* sama dengan frekuensi kerja menghasilkan nilai BER sebesar 0.4953 sedangkan untuk  $P_{tx} = -19$  dBm sebesar 0.4892. Hasil pengujian menunjukkan bahwa saat frekuensi *jamming* berkerja pada frekuensi kerja akan berpengaruh besar terhadap nilai BER yang dihasilkan baik pada *Multi Tone Jamming* maupun pada *Single Tone Jamming*. Saat frekuensi kerjanya di pengaruhi oleh frekuensi *jamming* akan menghasilkan nilai BER akan semakin besar. Begitu juga terhadap nilai daya *jamming* yang digunakan, akan menghasilkan nilai BER yang

semakin besar jika daya *jamming* yang digunakan semakin besar juga. Hasil  $P_{tx} = -23$  dBm untuk hasil sistem FHSS dan 4-ary FSK saat frekuensi *jamming* sama dan tidak sama dengan frekuensi kerja tidak dipengaruhi oleh gangguan *Multi Tone Jamming* pada daya *jamming* sebesar 30% sampai dengan 10%.



**Gambar 4. 6** Grafik BER vs Daya Sinyal *Jamming* pada *Multi Tone Jamming* saat  $P_{tx} = -23$  dBm

Berdasarkan hasil pengujian yang dilakukan pada sistem FHSS ini lebih baik dari pada sistem 4-ary FSK, untuk nilai sistem FHSS yang menggunakan gangguan *Single Tone Jamming* menghasilkan nilai BER yang lebih baik dari pada menggunakan gangguan *Multi Tone Jamming*. Hal ini karena, pada *Multi Tone Jamming* menggunakan dua frekuensi *jamming*. Dalam hasil yang didapat, nilai daya *jamming* juga berpengaruh terhadap nilai BER yang dihasilkan, apabila semakin besar nilai daya *jamming* maka akan menghasilkan nilai BER yang semakin besar juga dan begitu sebaliknya untuk nilai daya *jamming* yang digunakan kecil. Dalam pengujian ini, sistem FHSS tidak mengalami gangguan *Single Tone Jamming* saat frekuensi *jamming* sama dengan frekuensi pada daya *jamming* 75% ke bawah untuk nilai  $P_{tx} = -19$  dBm

dan saat  $P_{tx} = -23$  dBm tidak terpengaruh pada daya *jamming* 50% ke bawah. Sedangkan untuk *Single Tone Jamming* pada frekuensi *jamming* tidak sama dengan frekuensi kerja tidak mengalami gangguan saat daya *jamming* 100% ke bawah untuk nilai  $P_{tx} = -19$  dBm dan  $P_{tx} = -23$  dBm pada daya *jamming* 75% ke bawah. Hasil pengujian dari gangguan *Multi Tone Jamming* pada frekuensi *jamming* sama dengan frekuensi kerja tidak berpengaruh terhadap gangguan jamming untuk  $P_{tx} = -19$  dBm dan  $P_{tx} = -23$  dBm pada daya *jamming* 30% ke bawah. *Multi Tone Jamming* untuk frekuensi *jamming* tidak sama dengan frekuensi kerja pada  $P_{tx} = -19$  dBm tidak terpengaruh gangguan *jamming* saat daya *jamming* 75% ke bawah, sedangkan  $P_{tx} = -23$  dBm pada daya *jamming* 30% ke bawah. Untuk nilai  $P_{tx}$  yang digunakan semakin besar maka menghasilkan nilai BER yang semakin kecil. Jadi dari hasil pengujian ini yang dapat mempengaruhi nilai BER pada sistem FHSS ini adalah daya *jamming* dan nilai daya pancar yang digunakan.





## RIWAYAT HIDUP PENULIS



**Rini Nurpita Sari** dilahirkan di Padang, pada tanggal 10 Oktober 1990 merupakan putri pertama dari dua bersaudara dari pasangan Bapak Karnalis dan Ibu Nurkayah. Penulis menamatkan sekolah di SDN 17 Rawang pada tahun 2002. Kemudian masuk ke SMPN 1 Ulakan Tapakis, tamat tahun 2005, dan melanjutkan di SMAN 1 Ulakan Tapakis pada tahun yang sama. Tahun 2008, penulis melanjutkan pendidikan di Program Studi D3 Teknik Telekomunikasi, Jurusan Teknik Elektro, Politeknik Negeri Padang dan tamat pada tahun 2011. Setelah lulus D3 penulis berkesempatan merasakan dunia kerja di PT Hutchinson CP Telecommunication sebagai *Call Center* selama 5 bulan (Agustus 2012-Desember 2012). Selanjutnya penulis melanjutkan jenjang pendidikan S1 program Lintas Jalur di Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya pada akhir tahun 2012. Penulis memilih bidang studi Telekomunikasi Multimedia dan mengambil topik Tugas Akhir tentang '*Frequency Hopping Spread Spectrum*' di Laboratorium Komunikasi Multimedia.

E-mail : rininurpitasari@gmail.com  
Facebook : <https://www.facebook.com/rininurpitasari>  
Twitter : @Rininutpita